Micro Focus Security ArcSight SOAR 3.2

Software Version: 3.2

Integration Guides

Document Release Date: February 2022
Software Release Date: February 2022

MICRO FOCUS®

Legal Notices

Micro Focus
The Lawn
22-30 Old Bath Road
Newbury, Berkshire RG14 1QN
UK

https://www.microfocus.com

Copyright Notice

Trademark Notices

Adobe™ is a trademark of Adobe Systems Incorporated.

Microsoft® and Windows® are U.S. registered trademarks of Microsoft Corporation.

UNIX® is a registered trademark of The Open Group.

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number
- Document Release Date, which changes each time the document is updated
- Software Release Date, which indicates the release date of this version of the software

To check for recent updates or to verify that you are using the most recent edition of a document, go to:

https://www.microfocus.com/support-and-services/documentation

Contents

Integration Guide for AlientVault OTX

Integration Overview

AlienVault OTX is an open threat exchange platform supported by AlienVault and the community.

Adding a new line to Test.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with AlienVault OTX:

- IP Indicator
- Hash Indicator
- URL Indicator
- Domain Indicator
- Hostname Indicator

Use Case: Enrichment of artifacts detected in the organization

SOAR, when integrated with AlienValut OTX, can search for an artifact and gather information such as related threats and recent detections. This information may lead the

investigation into a different path, and analysts can investigate and root out malicious activities in their networks.

This integration can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- SOAR connects to AlienVault OTX API via HTTPS. Typically it runs on 443/tcp port. So access to this service is required.
- A user account is required for SOAR to connect to AlienVault OTX. It can be created from the following link:

  https://otx.alienvault.com

Configuring AlienVault OTX

- AlienVault OTX requires an API key for access. Users can retrieve it from

https://otx.alienvault.com/api after logging in with a valid credential.

Configuring SOAR

1. Click Configuration > Credentials > Create Credential
2. Fill in the Credential Editor form with the following information:

   a. **Type:** Internal Credential

   - **Name**: Display name of credential set (i.e., AlienVault OTX Credentials)
   - **Username**: Empty
   - **Password**: Empty
   - **Private Key**: API Key retrieved from the AlienVault OTX
3. Click Configuration > Integrations > Create Integration
4. Fill in the configuration form with the following information:

   - **Name:** Display name of AlienVault OTX integration on SOAR
   - **Type:** AlienVault OTX
   - **Address:** Address of the cloud service is standard: https://otx.alienvault.com

- **Configuration:** You need to specify the following configuration parameters

```
# Integration ID of the proxy integration to use when co
nnecting to current
# integration.
# If not provided, SOAR will try to use a direct connection.
#proxy.id=123
#Max count of fetching NIDS list for IP Indicator enrichment
#If not provided, SOAR will fetch last 10 NIDS(s)
#ip.indicator.nids.list.entry.count=10
#Max count of fetching URL list for IP Indicator enrichment
#If not provided, SOAR will fetch last 50 URL(s)
#ip.indicator.url.list.entry.count=50
#Max count of fetching URL list for Domain Indicator enrichment
#If not provided, SOAR will fetch last 50 URL(s)
#domain.indicator.url.list.entry.count=50
#Max count of fetching Malware list for Hostname Indicator
enrichment
#If not provided, SOAR will fetch last 50 Malware(s)
#hostname.indicator.malware.list.entry.count=50
#Max count of fetching URL list for Hostname Indicator enrichment
#If not provided, SOAR will fetch last 50 URL(s)
#hostname.indicator.url.list.entry.count=50
# configure how far (in minutes) into the past this enrichment
will look.
#cache.reusing.duration=20
```

- **Credential**: Name of the credential set you've just created on step 2. (i.e., AlienVault OTX Credentials).

- **Trust Invalid SSL Certificates:** Select this if Engine's certificate is self-signed or not recognized by browsers. Not selected.

- **Require Approval From:** Select user(s) from list to ask her/his approval before executing enrichments on this integration.

- **Notify**: Select user(s) from the list to notify when SOAR performs an enrichment on this integration.

5. Click Save to complete integration.

6. When you click the Test button the following pop up should be displayed if your credentials and address are valid.

Additional Notes

- AlienVault OTX integration on SOAR is an Advanced Action Script, and the content of the default script is accessible under Configuration > Customization Library.

- While defining the integration for the first time, you will encounter the following warning message, which is expected behavior for this type of integration.
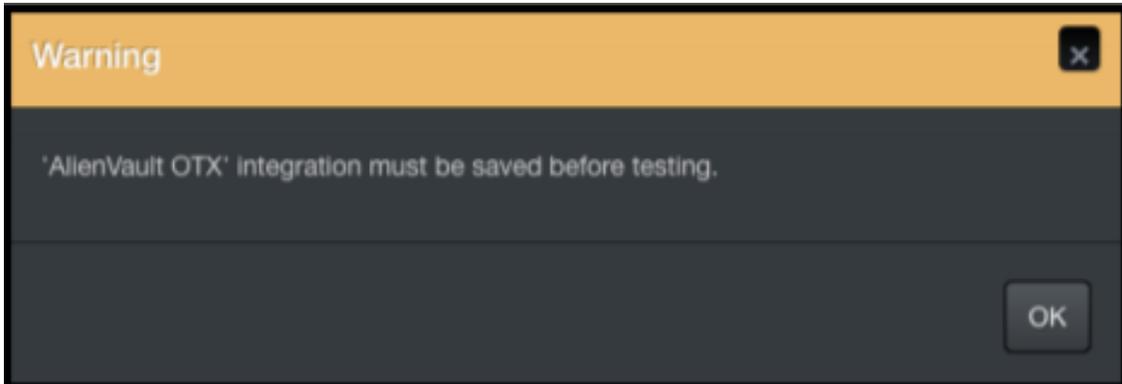
Integration Guide for Amazon EC2

Integration Overview

Amazon EC2 (Elastic Compute Cloud) forms a central part of Amazon.com's cloud-computing platform, Amazon Web Services, by allowing users to establish virtual networks and rent virtual computers on which they can run their own applications. Amazon EC2 REST-API supports the following Amazon Web Services:

- Amazon EC2
- Amazon EBS
- Amazon VPC
- AWS VPN

Please note that this integration is in Beta.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Amazon EC2:

- Add Network ACL Entry (VPC)
- Delete Network ACL Entry (VPC)

Use Case: Blocking Attackers

SOAR when integrated with Amazon EC2, blocks the attacker's IP addresses while responding to a cyber-attack. The blocking can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- SOAR connects to Amazon EC2 API via HTTPS. Access to https://ec2.amazonaws.com (443/tcp port) is required.
- AWS Access Key and AWS Access Key Secret are required for SOAR to connect

Amazon Web Services.

Configuring on Amazon AWS

1. Log in to Amazon Console (https://aws.amazon.com). Navigate to My Security Credentials, and select Identity Access Management (IAM) service:
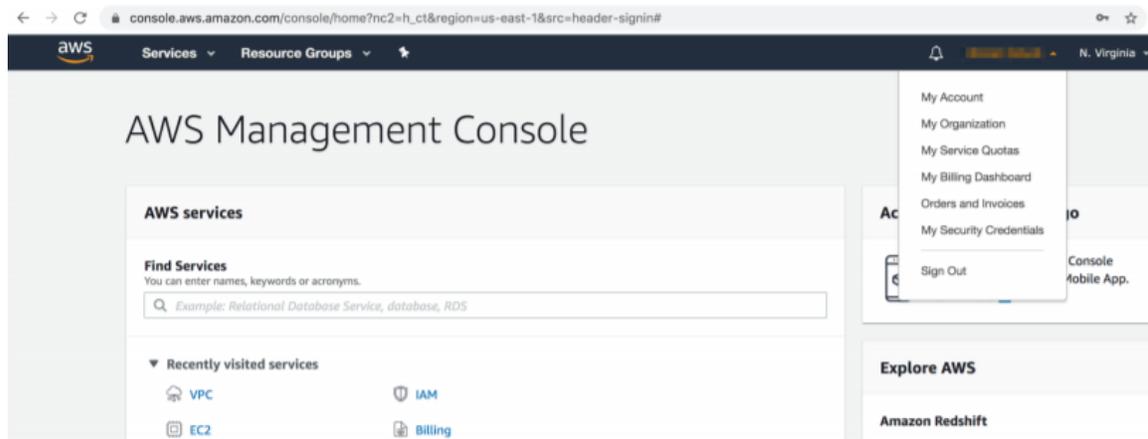
2. To add an IAM(identity and access management) user, click Access Management > Users > Add User. While adding new user account, it is important to select Access Type as Programmatic Access.
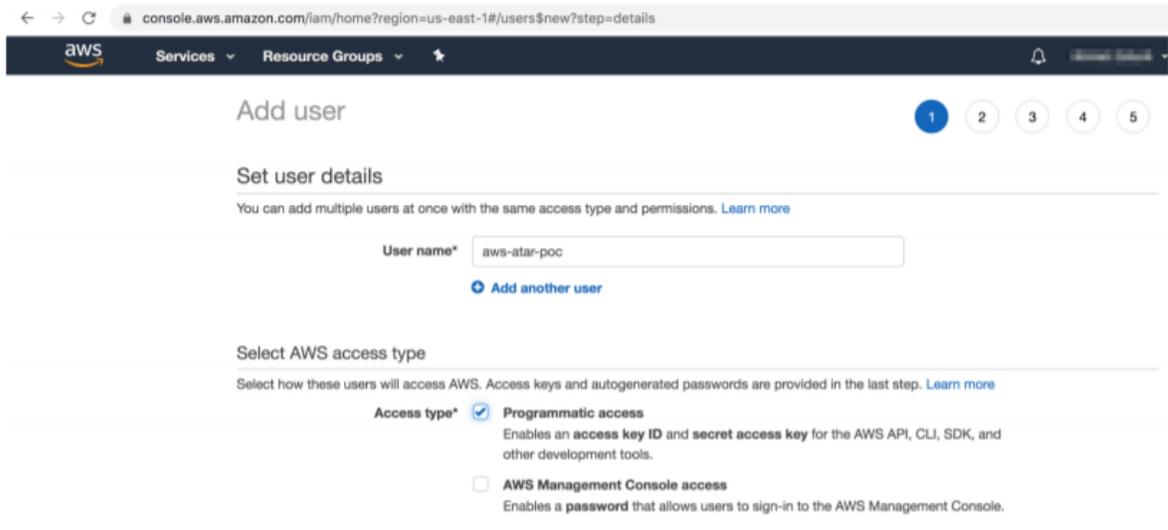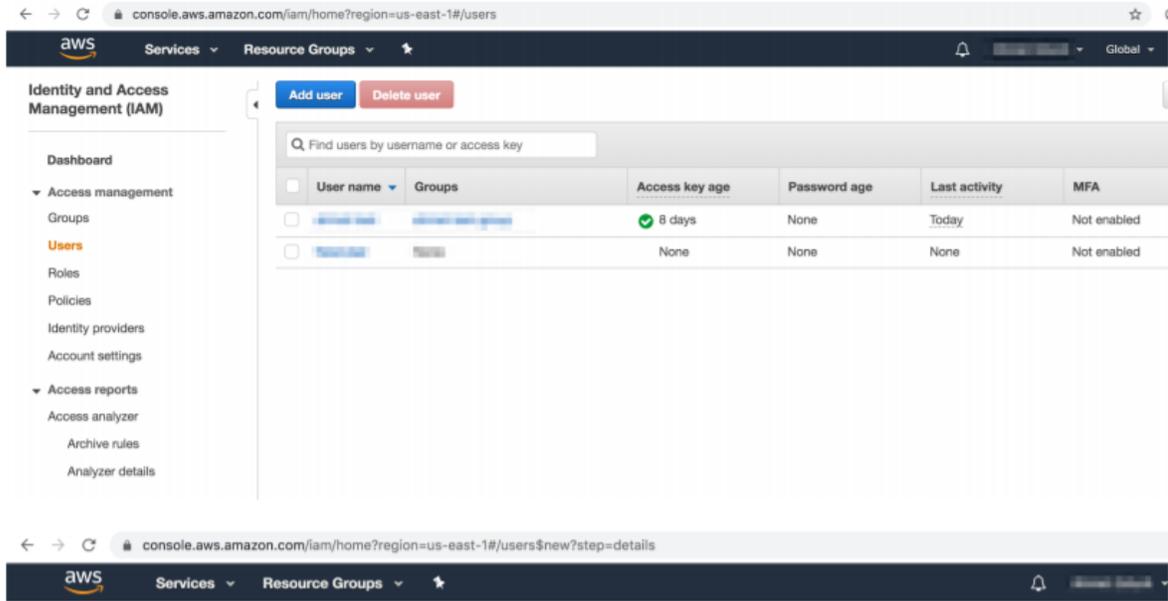
3. You can skip the next steps in the Add User process until Access Key and Access Key Secret are displayed.

> Note: Download the credentials as the Access Key Secret is never displayed post this step.
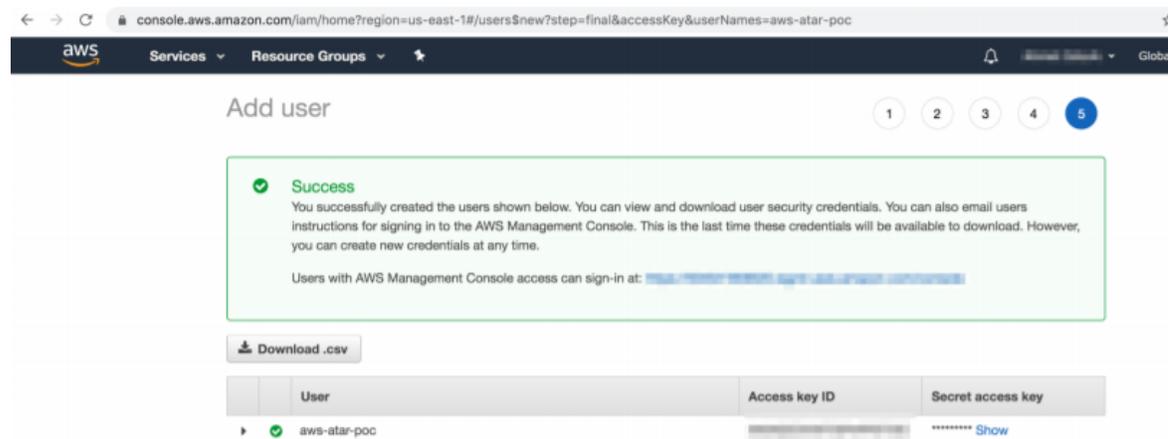
4. To arrange access policy, click > Access Management > Policies, and search for the required policy in previously defined policies list.

For example, the following image shows the policy AmazonVPCFullAccess.
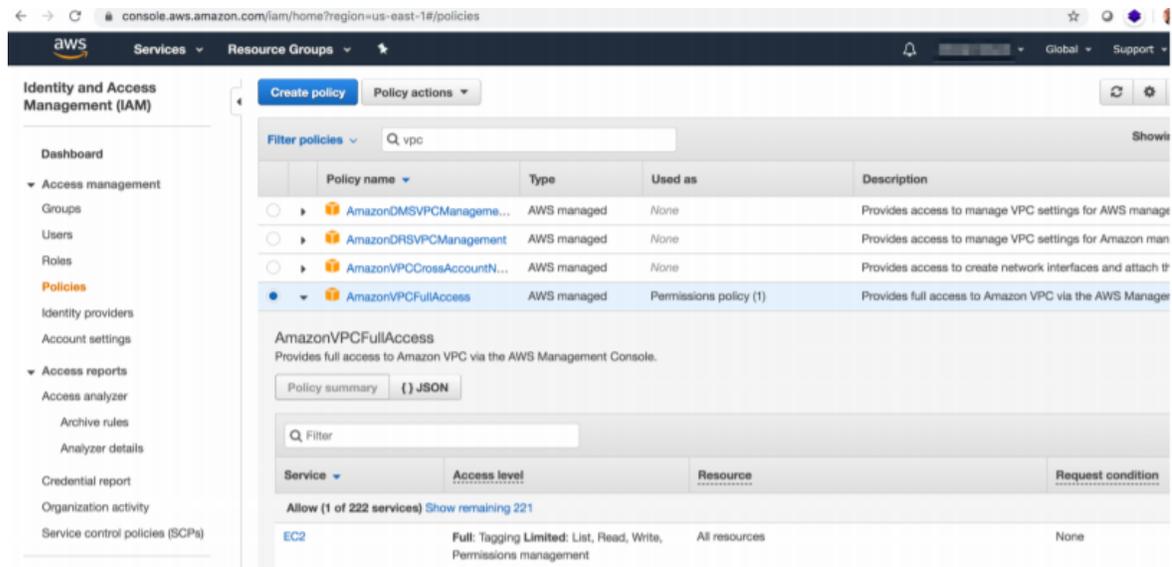
5. Select AmazonVPCFullAccess and open the Policy Summary.

a. **Click Policy Usage** > **Attach**.

b. In the Attach Policy menu, select the user that you have created in the previous steps, from the available users list in the system.
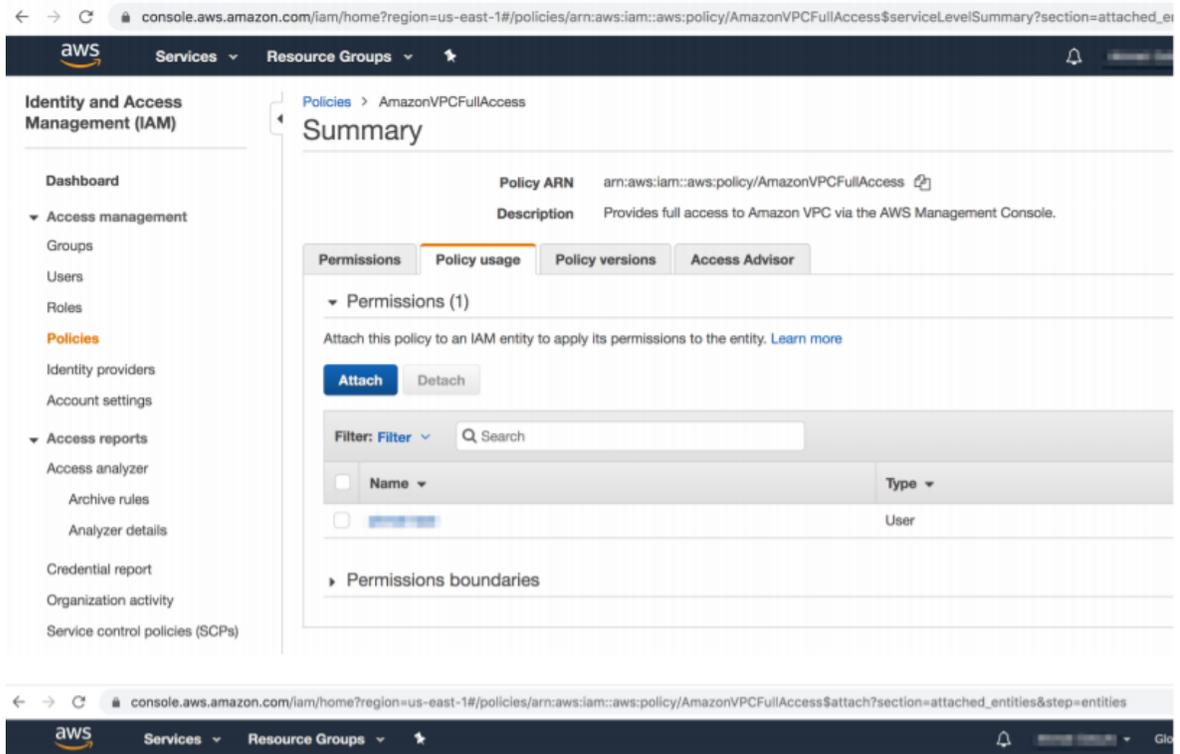
6. You can verify if the permission is successful for the user account that you've created on the Policy Usage page.
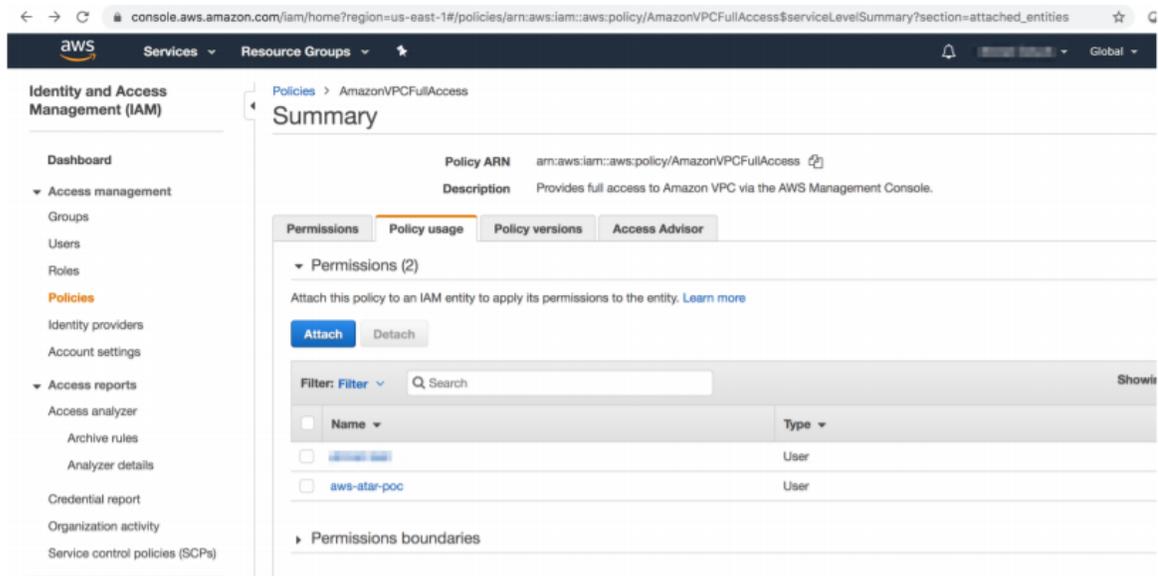
Configuring on SOAR

1. Click **Configuration** > **Credentials** > **Create Credential.**

2. Fill the Credential Editor form with the following information:

   a. Internal Credential:

   • **Type:** Internal credential

   • **Name:** Display name of credential set (i.e., Amazon AWS Credentials)

   • **Username:** Access Key of IAM user you have created

   • **Password:** Secret of Access Key of IAM user you have created

   • **Private Key:** Empty

   b. Credential Store:

   • **Type:** External credential

   • **Name:** Name of the credential with full path of the safe on store

3. Click Configuration > Integrations > Create Integration. Fill the Configuration form with the following information:

   • **Name:** Display name of Amazon EC2 integration on SOAR

   • **Type:** Amazon EC2

   • **Address:** Address of the integration (https://ec2.amazonaws.com)

   • **Configuration:** You need to specify the following configuration parameters

- **Credential:** Name of the credential set you have just created on step 2. (i.e., Amazon AWS Credentials)

- **Trust Invalid SSL Certificates:** No need to select

- **Require Approval From:** Select user(s) from list to ask her/his approval before executing actions on this integration

- **Notify:** Select user(s) from the list to notify when SOAR performs an action on integration

4. Click Save to complete integration.
5. Click the Test button. The following pop up will be displayed if your credential and address are valid.

Additional Notes

- Amazon EC2 integration on SOAR is an Advanced Script, and the content of the default script is accessible under **Configuration** > **Customization** Library.

- While defining the integration for the first time, you might encounter the following warning message, which is the expected behavior for this type of integration.
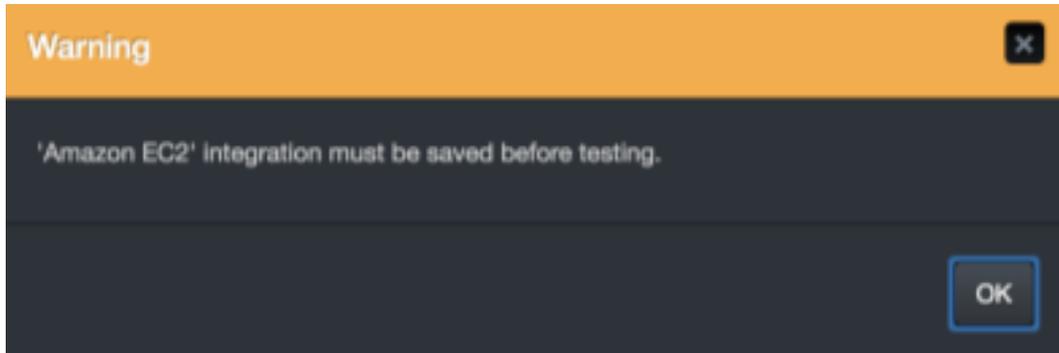
Integration Guide for Amazon IAM

Integration Overview

 Amazon AWS Identity and Access Management (IAM) enables you to manage access to AWS services and resources securely. Using IAM, you can create and manage AWS users and groups, and use permissions to allow and deny their access to AWS resources.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with amazon IAM:

- Add User to Group
- Create Group
- Deactivate MFA Device
- Delete Access Key
- Delete All Access Keys
- Delete All SSH Public Keys
- Delete All Service Specific Credentials
- Delete Group
- Delete Login Profile
- Delete SSH Public Key
- Delete Service Specific Credential
- Delete User Policy
- Delete Virtual MFA Device
- Detach User Policy
- Get Access Key Last Used
- Get Group (List Group Members)
- Get Policy
- Get User Policy
- Get User
- List Access Keys
- List Attached User Policies

- List Entities for Policy
- List Groups
- List Groups for User
- List MFA Devices
- List SSH Public Keys
- List Service Specific Credentials
- List User Policies
- List User Tags
- List Users
- Remove User from Group

Configuration

Prerequisites

- You must have access to HTTPS as the ArcSight SOAR connects to amazon iam API through this service.
- Access key is required to access this service.

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor** form:

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example,Amazon IAM Credential).	Empty	Access Key	Secret Key

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Specify the following parameter values in the **Configuration** form.

Parameter	Value
Name	Display name of the integration.
Type	Amazon IAM
Address	Address of the integration (the format must be https://iam.amazonaws.com).

Parameter	Value
Configuration	Specify the following configuration parameters: | **proxy.id**	**ID of the proxy integration if you access amazon web services through a web proxy device. For example: proxy.id = 12345** .	
Credential	Credential that has been defined for this integration in the **Credentials** menu.
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Save** to save the integration definition.

6. Navigate to **Configuration**>**Customization Library** and edit **Amazon IAM Advanced Action Script Default Template**.

7. Select the integration that you have added in the **Integrations** menu.

8. Click **Save** to complete the integration.

9. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Add User to Group**

   Action capability for adding a user to given group.

   - Rollback: Yes

   - Duplicate Control: No

   The following table presents the **Add User to Group** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is **no-rollback**.	N/A	N/A	No

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
User	Username to be added to group	Username Keyword Unknown	Yes	Yes
Group Name	Target group Name	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

2. **Create Group**

Action capability for creating a user group.

- Rollback: No

- Duplicate Control: False

The following table presents the**Create Group** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Path Prefix	Path where the group is created.	String	No	Yes
Group Name	Target group Name	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

3. **Deactivate MFA**

Action capability for deactivating user's multi factor authentication device.

- Rollback: No

- Duplicate Control: Yes

The following table presents the **Deactivate MFA** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes
Serial Number	MFA Device's serial number	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

4. **Delete Access Key**

   Action capability for deleting user's access key.

   - Rollback: No

   - Duplicate Control: Yes

   The following table presents the **Delete Access Key** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes
Access Key ID	Access Key ID	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

5. **Delete All Access Keys**

   Action capability for deleting user's all access keys.

   - Rollback: No

   - Duplicate Control: No

   The following table presents the **Delete All Access Keys** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes

Output:

Case Scope: No

Human Readable Output: No

6. **Delete All Service Specific Credentials**

   Action capability for deleting user's all service specific credentials.

   • Rollback: No

   • Duplicate Control: No

   The following table presents the **Delete All Service Specific Credentials** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

7. **Delete All SSH Public Keys**

   Action capability for deleting user's all SSH public keys.

   • Rollback: No

   • Duplicate Control: No

   The following table presents the **Delete All SSH Public Keys** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User		Username Keyword Unknown	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

8. **Delete Group**

   Action capability for deleting group.

   - Rollback: No

   - Duplicate Control: No

   The following table presents the **Delete Group** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Group Name	Group name to be deleted	String	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

9. **Delete Login Profile**

   Action capability for deleting user's login profile.

   - Rollback: No

   - Duplicate Control: No

   The following table presents the **Delete Login Profile** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

10. **Delete Service Specific Credential**

    Action capability for deleting user's service specific credential.

    • Rollback: No

    • Duplicate Control: Yes

    The following table presents the **Delete Service Specific Credential** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes
Credential ID	Service specific credential Id to be deleted	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

11. **Delete SSH Public Key**

    Action capability for deleting user's SSH public key.

    The following table presents the **Delete SSH Public Key** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes
SSH Public Key Id	SSH Public Key Id to be deleted.	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

12. **Delete User Policy**

Action capability for deleting user policy.

- Rollback: No

- Duplicate Control: No

The following table presents the **Delete User Policy** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes
Policy Name	Policy to be deleted.	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: No/A

13. **Delete Virtual MFA Device**

Action capability for deleting virtual multi factor authentication device.

- Rollback: No

- Duplicate Control: Yes

The following table presents the **Delete Virtual MFA Device** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Serial Number	Serial number of MFA device to be deleted.	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

14. **Detach User Policy**

Action capability for detaching policy from user.

- Rollback: No

- Duplicate Control: No

The following table presents the **Detach User Policy** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes
Policy arn	Policy to be detached.	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

15. **Get Access Key Last Used**

Enrichment capability for retrieving last used information for access key.

The following table presents **Get Access Key Last Used** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Access Key ID	Key ID to be queried .	String	No	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

User Name	Service	Region	Last Used Date
matt-acg	iam	us-east-1	1634811000

16. **Get Group**

Enrichment capability for retrieving list of group members.

The following table presents the **Get Group** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Group Name	Group Name	String	No	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

User Name	User Id	Arn	Path
iamdev		arn:aws:iam::462521641599:user/iamdev	/
iamdev2		arn:aws:iam::462521641599:user/iamdev2	/

17. **Get Policy**

Enrichment capability for retrieving policy information.

The following table presents the **Get Policy** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Policy arn	Policy arn.	String	No	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

18. **Get User**

   Enrichment capability for retrieving user details.

   The following table presents the **Get User** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

19. **Get User Policy**

Enrichment capability for adding a user to given group.

The following table presents the **Get User Policy** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username to be added to group	Username Keyword Unknown	Yes	Yes
Policy Name	Policy name	String	No	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

20. **List Access Keys**

Enrichment capability for listing user's access keys.

The following table presents the **List Access Keys** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

21. **List Attached User Policies**

Enrichment capability for listing attached user policies.

The following table presents the **List Attached User Policies** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output: No

22. **List Entities for User Policy**

Enrichment capability for listing entities for given user policy.

The following table presents the **List Entities for User Policy** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Policy Arn	Policy arn	String	No	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

Type	Entity
Policy User	iamdev
Policy User	iamdelete

23. **List Groups**

Enrichment capability for listing groups under given path prefix.

The following table presents the **List Groups** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Path Prefix	Path Prefix under groups to be listed.	String	No	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

Group Name	Group Id	Arn	Path	Create Date
Red Team	AGPAWXMDQFJ74NU000000	arn:aws:iam::462521641599:group/Devs/RedTeam	/Devs/	1622545035
Admins	AGPAWXMDQFJ7Y4I700000	arn:aws:iam::462521641599:group/Admins	/	1634813556

24. **List Groups for User**

Enrichment capability for listing user's groups.

The following table presents the **List Groups for User** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

25. **List MFA Devices**

Enrichment capability for listing user's MFA devices.

The following table presents the **List MFA Devices** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output: No

26. **List Service Specific Credentials**

Enrichment capability for listing user's service specific credentials.

The following table presents the **List Service Specific Credentials** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

27. **List SSH Public Keys**

Enrichment capability for listing user's SSH Public Keys..

The following table presents the **List SSH Public Keys** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username to be added to group	Username Keyword Unknown	Yes	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

28. **List Users**

Enrichment capability for listing users under the given path.

The following table presents the **List Uesrs** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Path Prefix	Path Prefix under users to be listed.	String	No	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

29. **List User Policies**

Enrichment capability for listing user's policies.

The following table presents the **List User Policies** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User		Username Keyword Unknown	Yes	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

30. **List User Tags**

    Enrichment capability for listing user's tags.

    The following table presents the **List User Tags** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	Username	Username Keyword Unknown	Yes	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output: No

Key	Value
Dept	Engineering
Role	UI Expert
Manager	Ahmet Ozturk

31. **Remove User from Group**

    Action capability for adding a user to given group.

    - Rollback: Yes

    - Duplicate Control: No

    The following table presents the **Add User to Group** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
User	Username to be added to group	Username Keyword Unknown	Yes	Yes
Group Name	Target group Name	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

Integration Guide for Amazon S3

Integration Overview

Amazon S3 service is offered by Amazon Web Services which provides object storage through a web service framework.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Amazon S3:

- Create Bucket
- Delete Bucket
- Download File From Bucket
- List Bucket Objects
- List Buckets
- Get Bucket Location

These capabilities can be performed automatically within a playbook or manually by an analyst.

Prerequisites

- You must have access to HTTPS as the ArcSight SOAR connects to Amazon S3 API through this service.
- Access Key ID and Secret Access Key is also required for integration.

Configuration

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor** form.

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, Amazon S3 Credential).		Access Key ID should be filled in this field.	Secret key should be filled in this field.

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Specify the following parameter values in the **Configuration** form.

Parameter	Value
Name	Display name of the integration.
Type	Amazon S3
Address	Address of the integration (the format must be https://s3.amazonaws.com).
Configuration	Specify the following configuration parameters: **proxy.id** — ID of the Proxy integration if you access Amazon S3 through a web proxy device. For example: proxy.id = 12345 . **region** — Default region name that has to be used while working on buckets. For example, proxy.id = 12345.
Credential	Credential that has been defined for this integration in the **Credentials** menu.
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Save** to save the integration definition.

6. Navigate to **Configuration**>**Customization Library** and edit **Amazon S3 Advanced Action Script Default Template**.

7. Select the integration that you have added in the **Integrations** menu.

8. Click **Save** to complete the integration.

9. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Create Bucket**
   Action capability for creating a bucket in Amazon S3.

   The following table presents the **Create Bucket** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Bucket Name	Name of the Amazon S3 Bucket that would be created.	String	N/A	Yes
Region	Region name of the bucket that would be created	List	N/A	No

Output:

Case Scope: N/A

Human Readable Output: N/A

2. **Delete Bucket**

Action capability for deleting a bucket in Amazon S3.

The following table presents the **Delete Bucket** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Bucket Name	Name of the Amazon S3 Bucket that would be deleted.	String	N/A	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

3. **Download File From Bucket**

Enrichment capability for downloading a file from bucket.

The following table presents the **Download File From Bucket** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Bucket Name	Name of the bucket that contains the file.	String	N/A	Yes
Key	Name of the file to be downloaded.	String	No	Yes
Do not Use Cache	SOAR does not use cached results if this box is checked.	Boolean	N/A	No
Region	Region name of the bucket that would be created	List	N/A	No

Output:

Case Scope:

Enrichment	Type	Category/ Value
Download File From Bucket	Any	File
Download File From Bucket	String	File Name
Download File From Bucket	MD5	#
Download File From Bucket	SHA1	#

Human Readable Output:

a few seconds ago - ▓▓ ▓▓ downloaded AMAZON_S3_2//DOWNLOAD_FILE_FROM_BUCKET—▓▓▓▓

4. **List Bucket Objects**

   Enrichment capability for listing bucket objects in Amazon S3.

   The following table presents the **List Bucket Objects** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Bucket Name	Name of the bucket that contains the file.	String	N/A	Yes
Do not Use Cache	SOAR does not use cached results if this box is checked.	Boolean	N/A	No
Region	Region name of the bucket that would be created	List	N/A	No

   Output:

   Case Scope: N/A

   Human Readable Output:

Key	Last Modified	Etag	Size In Bytes	Storage Class
▓▓▓▓	2021-09-28T10:08:11.000Z	▓▓▓▓	31	STANDARD

5. **List Buckets**

   Enrichment capability for listing a buckets in Amazon S3.

   The following table presents the **List Buckets**enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes

   Output:

   Case Scope: N/A

Human Readable Output:

6. **Get Bucket Location**

   Enrichment capability of getting region of the bucket.

   The following table presents the **List Buckets**enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Bucket Name	Name of the Bucket	String	No	Yes

   Output:

   Case Scope: N/A

   Human Readable Output:

Integration Guide for APIVoid

Integration Overview

APIVoid is an API service for threat analysis and threat detection and prevention.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with APIVoid:

- IP Reputation
- ThreatLog Domain Query
- Domain Reputation
- URL Screenshot
- URL Reputation
- Domain Age
- Site Trustworthiness
- Parked Domain
- URL Status
- HTTP Tracker
- Email Verify
- DNS Lookup
- DNS Propagation
- SSL Info
- URL to HTML
- URL to PDF

Prerequisites

- You must have the network access through APIVoid
- You must have the APIVoid API key.

Configuration

Configuring APIVoid

1. Register to **APIVoid**. After logging, the API key is available.
2. Click **My API Keys** and copy the API key.

Configuring SOAR

1. Click **Configuration** > **Integration** > **Create Integration**.
2. Click **Create**. In **Configuration Editor** specify following values to create a credential:

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, APIVoid Credential).			API Key that you copied from APIVoid portal.

3. Click **Save** to save the integration definition.
4. Navigate to **Configuration**>**Customization Library** and edit **APIVoid Advanced Action Script Default Template**.
5. Select the integration that you have added in the **Integrations** menu.
6. Click **Save** to complete the integration.
7. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

1. **IP Reputation**
   Enrichment capability for retrieving reputation value of given IP address.

   Following table presents the **IP reputation** enrichment capability details:

Capabilities

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
IP	IP address to retrieve reputation.	Network Address Host	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output:

2. **ThreatLog Domain Query**

Enrichment capability to query a domain for ThreatLog.

Following table presents the **ThreatLog Domian Query** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Domain	Host to query	HOST	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output

3. **Domain Reputation**

   Enrichment capability to retrieve Domain Reputation.

   Following table presents the **Domain Reputation** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Domain	Host to query	HOST	Yes	Yes

   Output:

   Case Scope: N/A
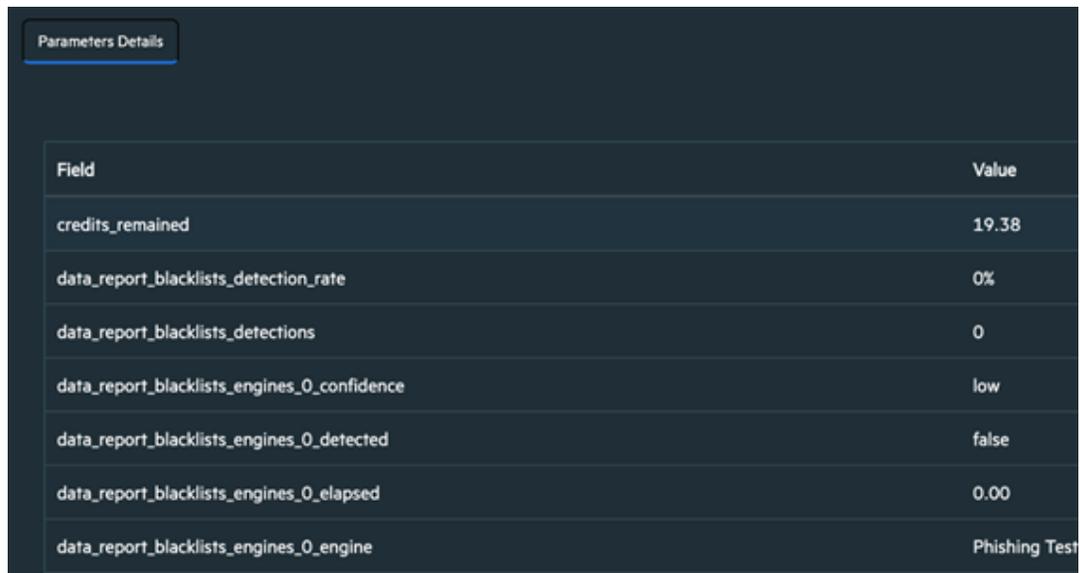
   Human Readable Output:

4. **URL Screenshot**

   Enrichment capability to take a screenshot for given URL by APIVoid.

   Following table presents the **URL Screenshot** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
URL	URL to take screenshot.	URL	Yes	Yes

   Output:

   Case Scope: N/A

   Human Readable Output:

5. **URL Reputation**

   Enrichment capability to retrieve URL reputation.

   Following table presents the **URL Reputation** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
URL	URL to retrieve reputation.	URL	Yes	Yes

   Output:

   Case Scope: N/A

   Human Readable Output:

6. **Domain Age**

   Enrichment capability to retrieve domain age information.

   Following table presents the **Domain Age** enrichment details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Domain	Domain to retrieve age information.	HOST	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output:

7. **Site Trustworthiness**

   Enrichment capability to retrieve site trustworthiness score / information

   Following table presents the **Site Trustworthiness** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Host	Host to retrieve site trustworthiness information.	HOST	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output:

8. **Parked Domain**

   Enrichment capability to retrieve information for parked domain.

   Following table presents the **Parked Domain** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Domain	Domain to retrieve information.	HOST	Yes	Yes

   Output:

   Case Scope: N/A

   Human Readable Output:

9. **URL Status**

   Enrichment capability to retrieve URL Status information.

   Following table presents the **URL Status** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
URL	URL to retrieve status.	URL	Yes	Yes

Output:
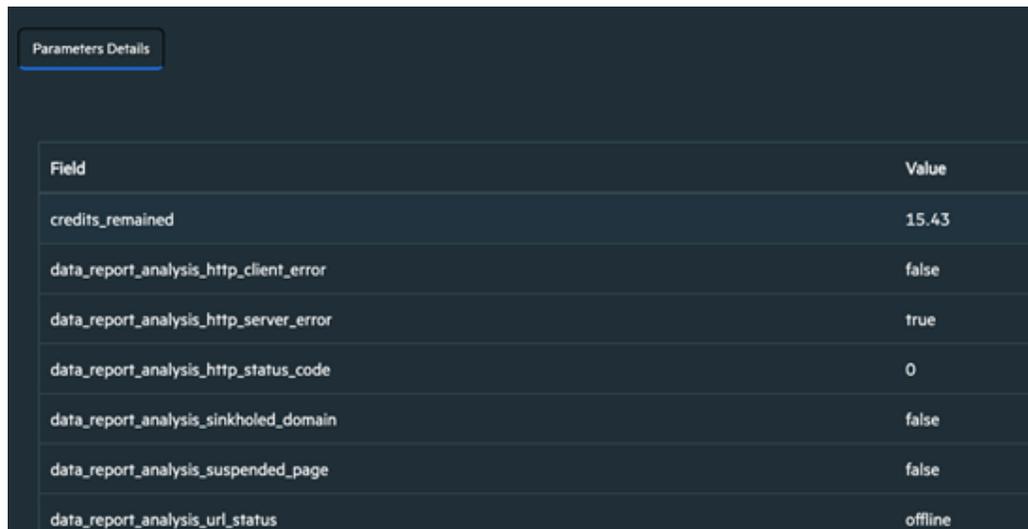
Case Scope: N/A

Human Readable Output:

10. **HTTP Tracker**

    Enrichment capability for tracking http requests per URL.

    Following table presents the **HTTP Tracker** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
URL	URL to track http requests.	HOST	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output:

11. **Email Verify**

Enrichment capability that verifies given E-mail address.

Following table presents the **Email Verify** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Email Address	Email Address to verify.	EMAIL_ ADDRESS	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output:

12. **DNS Lookup**

Enrichment capability to lookup for DNS per given host.

Following table presents the **DNS Lookup** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
HOST	Host or domain to lookup.	HOST	Yes	Yes

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Lookup Type	DNS Lookup type. Can be one of the following: "dns-a", "dns-aaaa", "dns-mx", "dns-ns", "dns-dmark", "dns-ptr", "dns-txt", "dns-any","dns-cname", "dns-soa", "dns-srv", "dns-caa" .	ENUM	No	Yes

Output:

Case Scope: N/A

Human Readable Output:

13. **DNS Propagation**

    Enrichment capability to check for DNS of the given host.

    Following table presents the **DNS Propagation** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Email Address	Host or domain to lookup.	HOST	Yes	Yes
Lookup Type	DNS Lookup type. Can be one of the following: "A", "AAAA", "NS", "MX", "TXT", "SRV", "PTR", "SOA", "CNAME", "SPF", "CAA" .	ENUM	No	Yes

Output:

Case Scope: N/A

Human Readable Output:

14. **SSL Info**

Enrichment capability to retrieve SSL information.

Following table presents the **SSL Info** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
HOST	Host or domain to lookup.	HOST	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output:

15. **URL to HTML**

Following table presents the **URL to HTML** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
URL	URL to retrieve HTML.	URL	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output:
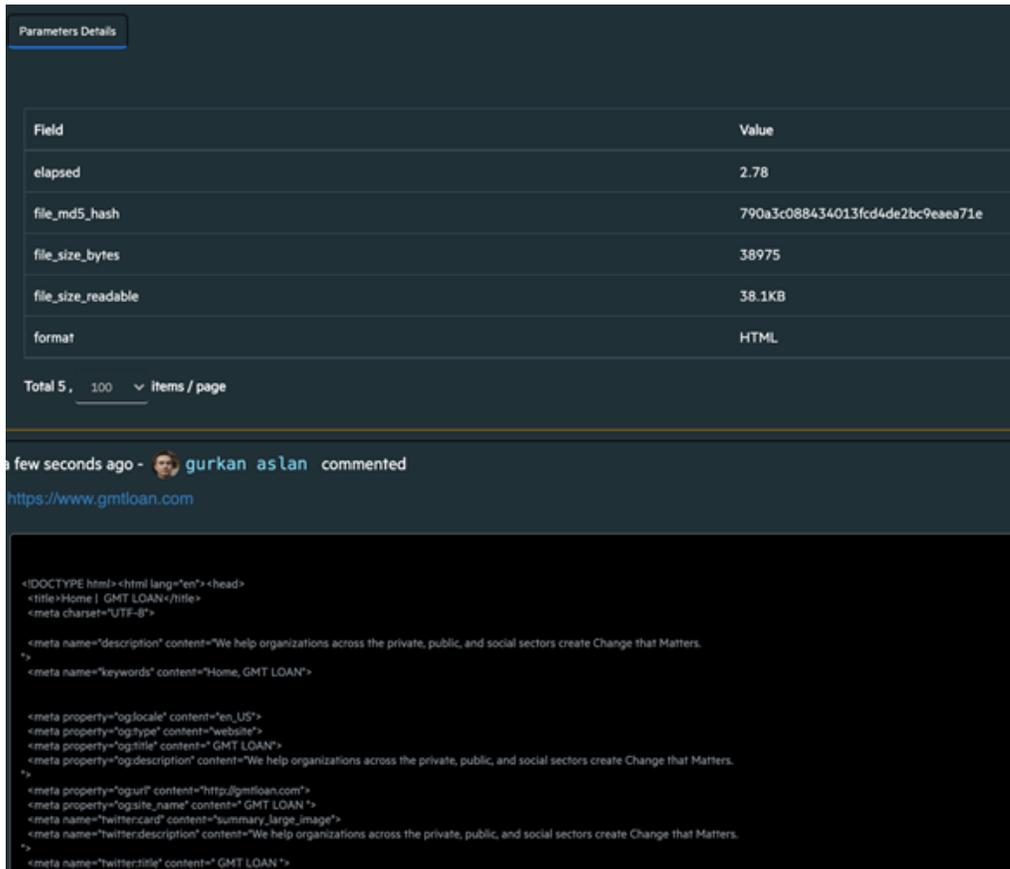
16. **URL to PDF**

Enrichment capability to retrieve PDF file from URL.

Following table presents the **URL to PDF** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
URL	URL to retrieve PDF.	URL	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output:

Field	Value
elapsed	1.95
file_md5_hash	ae5235c90cb5b93963c9e9474ccb7ae5
file_size_bytes	1626964
file_size_readable	1.6MB
format	PDF

Total 5 , 100 items / page

a few seconds ago - gurkan aslan commented

https://www.gmtloan.com

url.pdf (1589K)

Integration Guide for Anomali ThreatStream

Integration Overview

Anomali ThreatStream is a Threat Intelligence Platform that enables businesses to integrate security products and leverage threat data to defend against cyber threats.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Anomali ThreatStream:

- Domain Reputation
- Email Reputation
- File Reputation
- Get Incident Details
- Get Intelligence
- Get Investigation Details
- IP Reputation
- List Incidents
- List Investigations
- Report Indicator
- Create Investigation
- Close Investigation
- Update Investigation

Use Case: Investigating Phishing Campaigns

SOAR, when integrated with Anomali ThreatStream, helps campaigns that investigate and mitigate phishing. When a phishing report email comes from a user, SOAR extracts the indicators such as IP address, URLs and attachments in the message and creates an incident on the Incident Management Service Desk. SOAR then checks with Anomali ThreatStream, to know if this is a known attack and whether these indicators were previously analyzed.

This investigation can be either performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- SOAR connects to Anomali ThreatStream API via HTTPS. Access to https://api.threatstream.com/ **(443/tcp port)** is required.
- An API key is required for SOAR to connect to Anomali ThreatStream Service.

Configuring Anomali ThreatStream

1. Log in to https://ui.threatstream.com/.
2. Navigate to **Settings** > **My Profile** to get the API Key.

> **Note:** This key is required by SOAR to access the platform for queries.
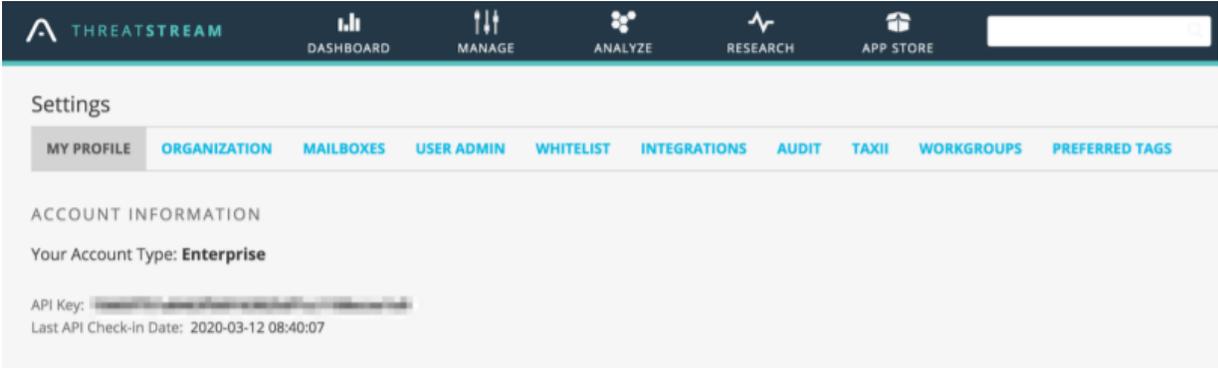
Configuring SOAR

1. **Configuration** > **Credentials** > **Create Credential.**
2. Fill the **Credential Editor** form with the following details:

    a. **Internal Credential:**

Parameter	Value
Type	Internal credential
Name	Display name of credential set (For example, Anomali ThreatStream Credentials)
Username	Your username on Anomali ThreatStream platform
Password	Empty
Private Key	API key you have obtained from Anamoli ThreatStream Platform

b. **Credential Store:**

Parameter	Value
Type	External credential
Name	Name of the credential with full path of the safe on store

3. **Configuration** > **Integrations** > **Create Integration**.

4. Fill the configuration form with the following parameter values:

Parameter	Value
Name	Display name of Anomali ThreatStream integration on SOAR
Type	Anomali ThreatStream
Address	Address of the integration (https://api.threatstream.com).
Configuration	You need to specify the following configuration parameters: ``` # Integration ID of the proxy integration to use when connecting to # current integration. # If not provided, ATAR will try to use a direct connection. #proxy.id=123 ```
Credential	Name of the credential set you have just created on step 2. (For example, Anomali ThreatStream Credentials)
Trust Invalid SSL Certificates	No selection required
Require Approval From	Select user(s) from list to ask her/his approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

.

5. Click **Save** to complete integration.
6. Click **Test** to test the integration.

Additoinal Notes

- Anomali ThreatStream integration on SOAR is an Advanced Script and content of the default script is accessible under **Configuration** > **Customization Library**.
- While defining the integration for the first time, you might encounter the following warning message, which is the expected behavior for this type of integration.

Warning ✕

'Anomali ThreatStream' integration must be saved before testing.

OK

Integration Guide for Arbor Networks APS

Integration Overview

Arbor Networks APS is an in-line Distributed Denial of Service(DDoS) protection solution.

Integration Capabilities

ArcSight has the following integration capabilities with Arbor Networks APS:

- Block IP
- Block access to IP

Use Case: Blocking malicious IP on peripheral

ArcSight SOAR integrates with Arbor Networks APS to block malicious IP addresses detected while responding to an incident. SOAR can block both the incoming and outgoing traffic either automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- SOAR connects to Arbor Networks APS' API via HTTPS. By default, the API interface works on **443/tcp port**. So access permission to this port is required.
- An API access token needs to be created for SOAR to connect to Arbor Networks APS.

Configuring Arbor Networks APS

1. Log in to Arbor Networks APS device.
2. Add a new API token.

```
admin@arbos: /# serv aaa local apitoken generate admin ATAR_INTEGRATION
   Added token: jwP9JcmZYz4I9QH0LpkDA_n5nj_DNHifc6Iwsq0P
```

> 🏠 **Note:** SOAR uses the generated token as the credential password and user name as admin.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.
2. Fill the **Credential Editor form** with the following parameter values:

   a. **Internal Credential:**

Parameter	Value
Type	Internal credential
Name	Display name of the credential set (For example, Arbor APS Credential)
Username	admin
Password	API Token you have created for SOAR on Arbor Networks APS device
Private Key	Empty

   b. **Credential Store:**

Parameter	Value
Type	Extrenal credential
Name	Name of the credential with pull path of the safe on store

3. **Configuration** > **Integrations** > **Create Integration**.
4. Fill the configuration form with the following parameter values:

Parameter	Value
Name	Display name of Arbor Networks APS integration on SOAR
Type	Arbor Networks APS
Address	Address of the integration (the format should be http(s]://1.1.1.1:1234 or http[s]://abc.example.com:1234)
Password	API Token you have created for SOAR on Arbor Networks APS device
Credential	Name of the credential set you have just created on step 2. (For example, Arbor APS Credential)

Parameter	Value
Trust Invalid SSL Certificates	Select this if device's certificate is self-signed or not recognized by browsers
Require Approval From	Select user(s) from list to ask her/his approval before executing actions on this integration
Notify	Select user(s) from the list to notify when ATAR performs an action on this ntegration

5. Click **Test**.The following pop up will be displayed if your credential and address are valid.

6. Click **Save** to complete integration.

Integration Guide for Bind RPZ DNS

Integration Overview

ArcSight SOAR uses BIND DNS servers to block malicious domains using incident scope.

Integration Capabilities

Action

- Block

Configuration

Prerequisites

- You must enable the DNS Zone Transfer on the server as SOAR uses DNS Zone Transfer Protocol to connect to the BIND DNS server.
- Remote Name Daemon Control (RNDC)

Configuring SOAR

1. To create the integration, navigate to **Configuration** > **Integrations**.
2. Specify the following parameter values in the **Integration Editor window**:

Parameter	Value
Name	Display name of the integration
Type	BIND RPZ DNS
Address	Address of the integration (the format must be 1.1.1.1).
Configuration	You must specify the following configuration parameters: • **ZONE**: Name of the RPZ configured on the BIND server • **BLOCK_IP**: IP address to which malicious domains need to be redirected • **TTL**: Time-to-live for the DNS record • **KEY_NAME**: Name of the RNDC key

Parameter	Value
Credential	Specify the Credential that was defined for this integration under the **Credentials** menu
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select users from list who can provide approval before executing action on this integration
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration

3. Click **Test**. The following pop up will be displayed if your credential and address are valid.
4. Click **Save** to complete integration.

Integration Guide for Carbon Black Response (EDR)

Integration Overview

Carbon Black Response (EDR)is a next-generation antivirus and end point detection response application. It's sophisticated detection combines custom and cloud-delivered threat intel, automated watchlists, and integrations with other platforms to efficiently scale hunt across the enterprise. It consolidates threat intelligence for your environment to automatically detect suspicious behavior.

Integration Capabilities

- Block Hash
- Unblock Hash
- Quarantine
- Unquarantine
- Computer Info
- Download Binary
- Get Binary Metadata
- List Process Connections
- Process Event Details
- Search Binaries
- Search Processes

Use Case: Investigating and Blocking Malware Spread

ArcSight SOAR integrates with Carbon Black Response (EDR), to help investigation and mitigation of malware attacks. When a suspicious file or malware is detected, SOAR lets you to search malware across endpoints, isolates PCs from network, and blocks relevant hashes. This investigation can either be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Access to port 443/tcp as SOAR connects to Carbon Black Response(EDR) API through HTTPS.
- An API key is required for SOAR to connect to Carbon Black Response(EDR).

Configuring Carbon Black Response(EDR)

1. Log in to Carbon Black Server.
2. Navigate to **User Profile** > **API Token** and make a note of the API key.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.
2. Specify the **Crenetial Editor** form with the following parameter values:
   a. **Internal credential:**

Parameter	Value
Type	Internal credential
Name	Display name of the credential set (For example, Carbon Black Credential)
Username	Empty
Password	Empty
Private Key	API Key obtained from Carbon Black Response (EDR)·

   b. **Credential Store:**

Parameter	Value
Type	External credential
Name	Name of the credential with full path of the safe on store.

3. Click **Configuration** > **Integrations** > **Create Migration**.

4. Specify the **Configuration form** with the following parameter values:

Parameter	Value
Name	Display name of Carbon Black Response (EDR) integration on SOAR
Type	Carbon Black Response
Address	Address of the integration (in the format: https://192.168.2.26)
Configuration	Specify the following configuration parameters: ``` # Integration ID of the proxy integration to use when connecting to # current integration. # If not provided, SOAR will try to use a direct connection. #proxy.id=123 ```
Credential	Name of the credential set created on step 2. (For example, Carbon Black Credentials)
Trust Invalid SSL Cerificates	Not Applicable
Require Approval From	Select users from list who can provide approval before executing actions on this integration.
Notify	Select users from the list to notify when SOAR performs an action on this integration

5. Click **Save** to complete integration.
6. Click **Test** to test the integration.

Additional Notes

- Carbon Black Response integration on SOAR is an Advanced Script, and the content of default script is accessible under **Configuration** > **Customization Library**.
- While defining the integration for the first time, you will encounter the following warning message, which is expected behavior for this type of integration.
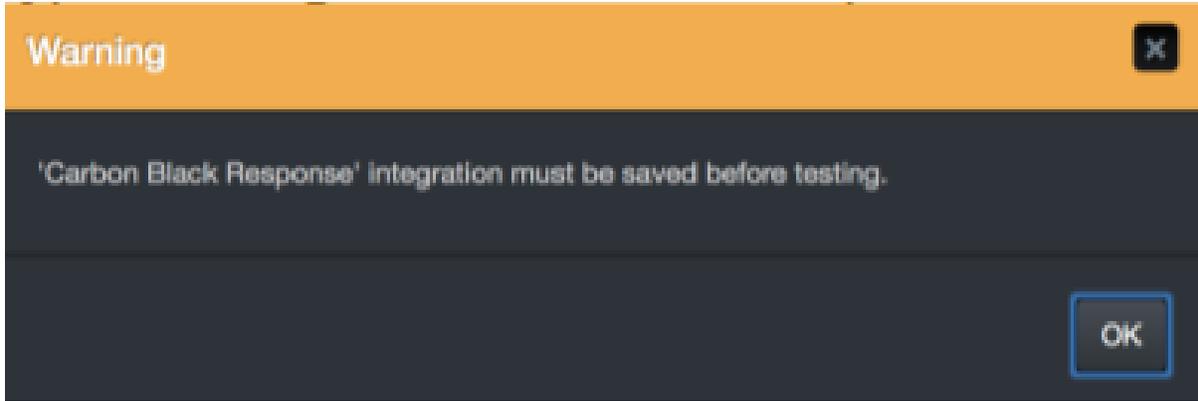
Integration Guide for Check Point R80

Integration Overview

Check Point R80 is an integrated solution for advanced threat prevention and security management.

This integration was tested with Check Point R80.20.

Integration Capabilities

- Block Email Sender
- Block Hash
- Block Host
- Block IP
- Block URL

Prerequisites

You must have access to HTTPS as the ArcSight SOAR connects to Check Point Smart Console API through this service.

Configuration

Configuring Check Point R80

1. Login to **Management Console** and navigate to **Manage & Settings > Blades > Management API Advanced Settings** and select **All IP addresses that can be used for GUI clients** in the Access Settings section.

2. Restart the API service by executing the following command in the command prompt:

   `api restart`

3. SOAR requires standard read/write access for the necessary policy and objects. To install policy automatically, the user must have the rights in its permission profile. You must

configure the required access rights for SOAR user as follows:

Type	Permission
Access Control	• Policy • Data Loss Prevention • Access Control Objects and Settings • Install Policy
Threat Prevention	• Policy Layers • Policy Exceptions • Profiles • Protections • Install Policy
Management	Management API Login
Others	Common Objects

4. Create an **Object Group** to be used by SOAR. The ArcSight SOAR adds the objects that you want to block in the Object Group.

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**.
2. Specify the following values in the **Credential Editor**:

Parameter	Value
Type	Internal Credential
Name	Display name of credential set, for example, Check Point R80 Credentials.
Username	User that you have created for SOAR on Check Point R80
Password	Password of the user you have created for SOAR on Check Point R80
Private Key	Empty

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Specify the following values in the **Configuration Form**:

Parameter	Value
Name	Display name of the integration.
Type	Check Point R80 Next Generation Firewall.
Address	Address of the integration (the format must be 10.0.0.1 or abc.example.com)

Parameter	Value
Configuration	Specify the following configuration parameters:

group.name	Object Group's name created in Check Point configuration steps. For example: `group.name = SOAR`
products	Possible values are AV (Anti Virus) for external threats and AB (Anti Bot) for internal threats. Please put "|" separator for more than one product. For example: `Product = AV|AB`
install.policy	If you would like to install policy automatically, set this variable true: `install.policy = true`
policy.package	Policy which SOAR installs on target systems. Required if install.policy is true. For example: `policy.package = standard`
targets	Name of the target gateways. Required if install.policy is true. Please use "|" as separator if you have more than one target. For example: `targets = CP_Cluster`
access	Required for blocking IP addresses on access policy. Required if install.policy is true. `access = true`
threat.prevention	Required for blocking indicators on Threat Prevention policy (Domain, Email, Hash, URL). Required if install.policy is true. `threat.prevention = true`
proxy.id	ID of the Proxy integration if you access Check Point R80 through a web proxy device. For example: `proxy.id = 12345`

Parameter	Value
Credentials	Credential that has been defined for this integration under the Credentials menu.
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select user(s) from list to ask her/his approval before executing actions on this integration
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration

5. Click **Show Additional Parameters** checkbox and select the frequency of policy install in **Maintenance** dropdown.

> As the firewall might get overloaded, in case of frequent attacks or misconfiguration, thus, SOAR does not install the policy after every action. Instead, you can define the frequency of the policy install in **Maintenance** menu by either selecting pre-defined values or by defining a custom Cron expression for scheduling. The ArcSight SOAR uses spring-framework's Cron expression format. For the format and similar example, refer to the Spring Framework-Cron Expression

6. Click **Test**. An **Integration Successful** message is displayed if your credential and address are valid.

7. Click **Save** to complete the integration.

Capabilities

1. **Block Email Sender**

   Action capability for blocking malicious email addresses.

   - Rollback: Yes

   - Duplicate Control: Yes

   > Only supported on AV product. AB product doesn't support this capability.

Input Parameter	Description	Type	Scope Restricted Yes/No	Required Yes/No
Integration	Name of the third party integration	Integration	N/A	Yes
Rollback	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
Email Address	Email address to be blocked	Email Address	Yes	Yes

   Output:

   Case Scope: N/A

   Human Readable Output: N/A

2. **Block Hash**

   Action capability for blocking hash values of malicious files.

   - Rollback: Yes

   - Duplicate Control: Yes

> Only supported on AV product. AB product doesn't support this capability.

Input Parameter	Description	Type	Scope Restricted Yes/No	Required Yes/No
Integration	Name of the third party integration	Integration	N/A	Yes
Rollback	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
Hash	Hash to be blocked	Hash	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

3. **Block Host**

   Action capability for blocking malicious hosts.

   - Rollback: Yes

   - Duplicate Control: Yes

   > Only supported on AV product. AB product doesn't support this capability.

Input Parameter	Description	Type	Scope Restricted Yes/No	Required Yes/No
Integration	Name of the third party integration	Integration	N/A	Yes
Rollback	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
Host	Host to be blocked	Host (It is mentioned as domain object on Check Point)	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

4. **Block IP**

   Action capability for blocking malicious IP addresses.

- Rollback: Yes

- Duplicate Control: Yes

Only supported on AV product. AB product doesn't support this capability.

Input Parameter	Description	Type	Scope Restricted Yes/No	Required Yes/No
Integration	Name of the third party integration	Integration	N/A	Yes
Rollback	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
IP Address	IP address to be blocked	Network Address	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

5. **Block URL**

   Action capability for blocking URLs.

   - Rollback: Yes

   - Duplicate Control: Yes

   Only supported on AV product. AB product doesn't support this capability.

Input Parameter	Description	Type	Scope Restricted Yes/No	Required Yes/No
Integration	Name of the third party integration	Integration	N/A	Yes
Rollback	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
URL	URL to be blocked	URL	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

Integration Guide for Check Point SandBlast

Integration Overview

Check Point SandBlast provides advanced threat protection against known threats, zero-day malware, and sophisticated attacks.

Integration Capabilities

Threat Emulation capability prevents infections from undiscovered exploits, zero-day and targeted attacks by inspecting files, and running them in a virtual sandbox to discover malicious behavior.

ArcSight SOAR has the following integration capabilities with Check Point SandBlast:

- Threat Emulation & AV Scan

Use Case: Investigating suspicious file

With Check Point SandBlast integration, during the investigation of an incident, SOAR can send a suspicious file to Check Point SandBlast to emulate threats and run an anti virus scan for the file. This investigation can either be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Make sure you have access to 443/tcp port as SOAR connects to Check Point SandBlast's API through HTTPS. If cloud-based threat emulation service is used, the API interface works on https://te.checkpoint.com/api/bla/bla
- If a local gateway is used, typically access permission to 18194/tcp port is required.
- An API key is required for SOAR to connect to Check Point SandBlast.

Configuring Check Point SandBlast

1. If you are using cloud-based threat emulation service, contact Check Point to get the API key.

2. If you are using local gateway, the following link provides you with the document for creating API key:

   http://supportcontent.checkpoint.com/solutions?id=sk113599

Configuring SOAR

1. **Configuration** > **Integrations** > **Create Integration**.
2. Fill the **Credential Editor** form with the following parameter values:
   a. **Internal Credential:**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Check Point SandBlast Credential)
Username	Empty
Password	Empty
Private Key	API key you have created for SOAR on local gateway or you have obtained from Check Point.

   b. **Credential Store:**

Parameter	Value
Type	External credential
Name	Name of the credential with full path of the safe on store

3. **Configuration** > **Integrations** > **Create Integration.**
4. Fill the configuration form with the following parameter values:

Parameter	Value
Name	Display name of Check Point SandBlast integration on SOAR
Address	Address of the integration (the format must be https://192.168.1.1:18194 or https://te.checkpoint.com)
Credential	Name of the credential set you have just created on step 2. (For example, Check Point SandBlast Credential).
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or is not recognized by browsers.

Parameter	Value
Configuration	Specify the following configuration parameters: ```# Set local_instance true if you use local gateway.\nlocal_instance=false#\nconfigure how far (in minutes) into the past this enrichment will look.\ncache.reusing.duration=60\n# Set proxy id if necessary for SOAR to reach the SandBlast instance.\nproxy.id=123```
Require Approval Form	Select user(s) from list to ask her/his approval before executing actions on this s.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Test**. The following pop up will be displayed if your credential and address are valid.
6. Click **Save** to complete integration.

Integration Guide for CiscoASA Firewall

Cisco ASA is a security technology that combines firewall, antivirus , intrusion prevention and virtual private network (VPN) capabilities. It provides proactive threat defence and stops attacks before they spread in the network.

Integration Capabilities

- Block Host
- Block IP

Prerequisites

- You must have access to 443/tcp port for HTTPS as the ArcSight SOAR connects to Cisco ASA Firewall REST-API interface through this service.
- SOAR must have a user account to connect to Cisco ASA Firewall.

Configuration

Configuring Cisco ASA Firewall

1. Log in to **Cisco ASA Firewall** device command line console.
2. Create a user account with privilege level 15 as follows:

```
# configure terminal
```

```
# username soar password choose_a_complex_password privilege 15
```

3. Enable the **REST API** services by running the following commands:

```
# rest-api image
```

```
# rest-api agent
```

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor** form.

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, Cisco ASA Firewall Credential).	User you have created for SOAR on Cisco ASA Firewall.	Password of the user you have created for SOAR on Cisco ASA Firewall.	Empty.

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Specify the following parameter values in the **Configuration** form.

Parameter	Value
Name	Display name of the integration.
Type	Cisco ASA Firewall
Address	Address of the integration (the format should be https://10.0.0.1)
Configuration	Specify the following configuration parameters:

NETWORK_OBJECT_ GROUP_NAME_FOR_ IP	IP Object Group name used by SOAR. For example: NETWORK_OBJECT_GROUP_NAME_FOR_IP=SOAR_IP_LIST .
NETWORK_OBJECT_ GROUP_NAME_FOR_ DOMAIN	FQDN Object Group name used by SOAR. For example: NETWORK_OBJECT_GROUP_NAME_FOR_DOMAIN=SOAR_ DOMAIN_LIST.
proxy.id	ID of the Proxy integration if you access Cisco ASA Firewall through a web proxy device. For example: proxy.id = 12345

Parameter	Value
Credential	Credential that has been defined for this integration in the **Credentials** menu.
Trust Invalid SSL Certificates	Select this if firewall's web certificate is self-signed or is not recognized by browsers.
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Save** to save the integration definition.
6. Navigate to **Configuration**>**Customization Library** and edit **Cisco ASA Firewall Advanced Action Script Default Template.**

7. Select the integration that you have added in the **Integrations** menu.

8. Click **Save** to complete the integration.

Capabilities

1. **Block Host**
   Action capability for blocking malicious host.
   - Rollback: Yes
   - Duplicate Control: Yes

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the 3rd party integration	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback	N/A	N/A	No
FQDN	Host to be blocked	Host (It is written as domain object on Cisco ASA Firewall)	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

2. **Block IP**
   Action capability for blocking malicious IP addresses.
   - Rollback: Yes
   - Duplicate Control: Yes

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the 3rd party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
IP Address	IP address to be blocked	Network Address	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

Integration Guide for Cisco Firepower Management Center

Integration Overview

Cisco Firepower Management Center (formerly Sourcefire Firepower Management Center) is an administrative center node of the Firepower Threat Defense systems and manages critical Cisco network security solutions. It provides complete and unified management over firewalls, application control, intrusion prevention, URL filtering, and advanced malware protection.

This integration is tested with Cisco Firepower Management Center version 6.3.0 (build83).

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Cisco Firepower Management Center:

- Block IP
- Block URL

Prerequisites

You must have access to HTTPS as the ArcSight SOAR connects to Cisco Firepower Management Center REST API through this service.

Configuration

Configuring Cisco Firepower Management Center

1. Login to **Management Center** and navigate to **System** > **Configuration** > **REST API Preferences** and enable **REST API**.

2. Navigate to **System** > **Users** > **User Roles** and create a new role with the following permissions:

- **Object Manager>Modify Object Manager**

- **Deploy Configuration to Devices**

3. Navigate to **System** > **Users** > **Users** and create a new user account with user role that you have created in the previous step.

4. Navigate to **Objects** > **Object Management** and create two object groups with the following configurations.

Name	Description	Allow Overrides
SOAR_BLOCK_IP	Object Group for IPs blocked by ArcSight SOAR.	True
SOAR_BLOCK_URL	Object Group for URLs blocked by ArcSight SOAR.	True

> **Note**: You can use these object groups in required rules.

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor** form.

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, Cisco FMC Credential).	User you have created for SOAR on Cisco Firepower Management Center.	Password of the user that you have created for SOAR on Cisco Firepower Management Center.	

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Specify the following parameter values in the **Configuration** form.

Parameter	Value
Name	Display name of the integration.
Type	Cisco Firepower Management Center.
Address	Address of the integration (the format must be https://10.10.20.40).
Configuration	Specify the following configuration parameters:

proxy.id	ID of the Proxy integration if you access Cisco Firepower Management Center through a web proxy device. For example: proxy.id = 12345 .
network.object.group.name	Name of the object group SOAR adds IP addresses into. network.object.group.name = SOAR_BLOCK_IP .
url.object.group.name	Name of the object group SOAR adds IP addresses into. url.object.group.name=SOAR_BLOCK_URL.

Credential	Credential that has been defined for this integration under the **Credentials** menu.

Parameter	Value
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Save** to save the integration definition.

6. Navigate to **Configuration**>**Customization Library** and edit **Cisco Firepower Management Center Advanced Action Script Default Template**.

7. Select the integration that you have added to **Integrations** menu.

8. Click **Save** to complete the integration.

9. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Block IP**
   Action capability for adding an IP to given object group.

   • Rollback: Yes

   • Duplicate Control: No

   This table presents the **Block IP** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback	N/A	N/A	No
IP	IP address to be added to object group	Network Address	Yes	Yes

   Output:

   Case Scope: N/A

   Human Readable Output: N/A

2. **Block URL**
   Action capability for adding an URL to given object group.

- Rollback: Yes

- Duplicate Control: No

This table presents the **Block URL** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback	N/A	N/A	No
URL	URL to be added to object group	URL	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

Integration Guide for Cisco Identity Service Engine

Integration Overview

The Cisco Identity Services Engine (ISE) offers a network-based approach for adaptable, trusted access everywhere, based on the context. It provides intelligent, integrated protection through intent-based policy and compliance solutions.

Integration Capabilities

ArcSight SOAR has the following integration capability with Cisco Identity Services Engine:

Action:

- Block MAC Address

Configuration

Prerequisites

Make sure to check the following prerequisites:

- Current version of Cisco Identity Services Engine 2.3.0.238 as SOAR supports it.
- Access to 443/tcpport as SOAR connects to Identity Services Engine API through HTTPS.
- An user account for SOAR to connect to Identity Services Engine

Configuring Cisco Identity Services Engine

1. Create a user account and the user must be a member of MnT Admin.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**
2. Fill the **Credential Editor** form with following parameter values:

a. **Internal Credential:**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Cisco ISE credentials)
Username	User you have created for SOAR on Cisco Identity Services Engine
Password	Password of the user that you have created for SOAR on Cisco Identity Services Engine.
Private Key	Empty

b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store.

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Fill the configuration form with the following parameter values:

Parameter	Value
Name	Display name of Cisco Identity Services Engine integration on SOAR
Type	Cisco Identity Services Engine
Address	Address of the integration (the format must be https://192.168.2.3)
Credential	Name of the credential set you have just created on step 2 (For example, Cisco ISE Credentials)
Trust Invalid SSL Certificates	Select this if Firewall's certificate is self-signed or is not recognized by browsers
Configuration	You must specify the following configuration parameters. `serverHost =`
Require Approval From	Select user(s) from list to ask her/his approval before executing actions on this integration
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Test**. The following pop up will be displayed if your credential and address are valid.

6. Click **Save** to complete integration.

Integration Guide for Cisco Ironport Email Security

Integration Overview

Cisco Ironport Email Security is one of Cisco Ironport products to prevent phishing, business e-mail compromise, ransomeware and spam.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Cisco Ironport Email Security:

- Block sender IP/Host

- Block email that includes a keyword

- Block sender email

Use Case: Stopping phishing campaigns

With this integration, SOAR can block emails based on sender, IP address or a keyword while responding to cyber-attacks. Blocking can be either performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

Make sure to check the following prerequisites:

- Current version of Cisco Ironport Email Security 11.0.0-264 as SOAR supports it.

- Access to 22/tcp port as SOAR connects to Cisco Ironport Email Security via SSH.

- A user account for SOAR to connect to Cisco Ironport Email Security.

Configuring Cisco Ironport Email Security

1. To access the **Cisco Inroport Email Security resources**, create a user account with minimum **operator** role.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.
2. Fill the Credential Editor form with the following parameter values:
   a. **Internal Credential:**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Cisco Firepower Management Credentials)
Username	User you have created for SOAR on on Cisco Firepower Management Center
Password	Password of the user that you have created for SOAR on Cisco Firepower Management Center.
Private Key	Empty

   b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store.

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Fill the configuration form with the following parameter values:

Parameter	Value
Name	Display name of Cisco Ironport Email Security integration on SOAR
Type	Cisco Ironport Email Security
Address	Address of the integration (the format must be 192.168.200.43)
Credential	Name of the credential set you have just created on step 2 (For example, Cisco Ironport Credentials)
Require Approval From	Select user(s) from list to ask her/his approval before executing actions on this integration
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Save** to complete integration.
6. Click **Test** to test the integration.

Additional Notes

- Cisco Ironport Email Security integration on SOAR is an Advanced Action Script, and you can access the content of the default script under **Configuration** > **Customization Library**.

- While defining integration for the first time, you might encounter the following warning message, which is the expected behavior for this type of integration.
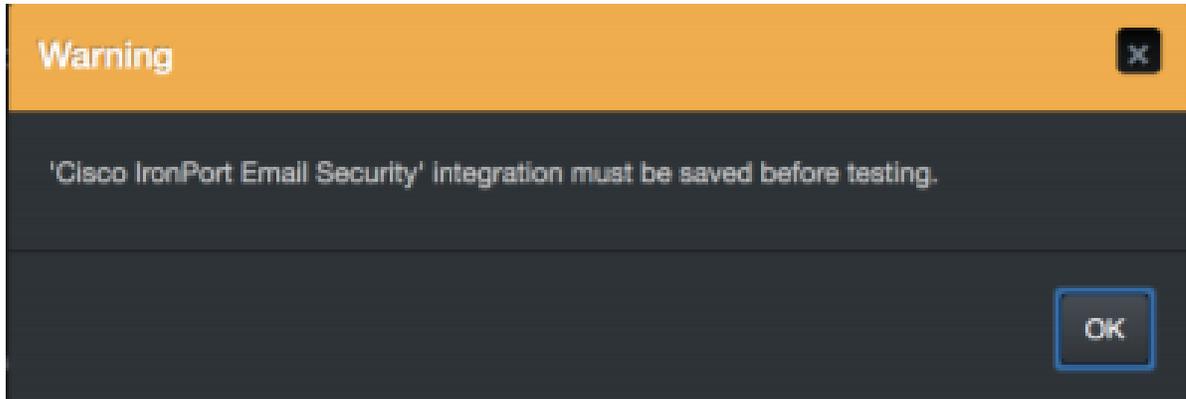
Integration Guide for Cyberark Central Credential Provider

Integration Overview

CyberArk Application Identity Manager is a central credential provider that stores passwords and other credentials used by systems, applications, and scripts by eliminating embedded credentials. SOAR might use encrypted credentials stored on its database and CyberArk AIM vault to connect to other systems and applications while investigating and responding to an incident.

Configuration

Prerequisites

- Make sure to check the access to CyberArk Application Identity Manager API as SOAR connects to it through HTTPS.
- Define **a new application for SOAR** on CyberArk's PVWA (Password Vault Web Access) Interface.

Configuring CyberArk Application Identity Manager

1. Log in to **Password Vault Web Access** interface as a user with **Manage Users** authorization permission.
2. Navigate to **Applications**and click **Add Application**.
3. Fill the Add Application form with the following parameter values:

Parameter	Value
Name	Specify SOAR as the unique name (ID) of the application.
Description	Specify a short description of the application (For example, Application for Automated Threat Analysis&Response)
Business Owner	Specify contact information about the application's Business owner
Location	Specify the location of the application in the Vault hierarchy. **Note:** If the location is not selected, the application gets added to the user location who creates it.

4. To specify unlimited number of machines and Windows OS users for a single application, select **Allow extended authentication restrictions**.

5. Navigate to **Allowed Machines** and specify the application's Allowed Machines.

> **Note:** This information enables the Credential Provider to check only applications that run from specified machines can access their passwords.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.

2. Fill the **Credential Editor** form with the following parameter values:

   a. **Internal Credential**:

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, CyberArk AIM Credential)
Username	Application Name you have created on CyberArk Password Vault Web Access
Password	Empty
Private Key	Empty

3. Click **Configurations** > **Integrations** > **Create Integration**.

4. Fill the **Configuration** form with the following details:

Parameter	Value
Name	Display name of CyberArk AIM integration on SOAR
Type	CyberArk Central Credential Provider
Address	Address of the integration (the format must be https://192.168.1.1:1234 or https://abc.example.com:1234)
Credential	Name of the credential set you have just created on step 2 (For example, CyberArk AIM Credential).
Trust Invalid SSL Certificates	Select this if device's certificate is self-signed or is not recognized by browsers
Require Approval From	Select user(s) from list to ask her/his approval before executing actions on this integration
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration

5. Click **Save** to complete integration.

6. Click **Test** to test the integration.

Additional Notes

Following are the steps to use CyberArk AIM as central credential store:

1. Navigate to **Configuraiton** > **Parameters**.

2. Modify the **ExternalCredentialStoreIntegrationID** parameter value to ID of the CyberArk AIM integration that you have defined in the above procedure.

3. To define the new name for a credential:

   a. Navigate to **Configuration** > **Credentials.**

   b. Select **External Credential** from the drop down and it automatically uses CyberArk AIM integration.

   > **Note:** The name of the credential must be the same as the account name defined in CyberArk. Make sure to follow the naming convention of SOAR as Safe and Folder separated by **|** character. Else, SOAR automatically searches all Safes for the given credential name.

Integration Guide for CYMRU Malware Hash Registry Query

Integration Overview

CYMRU is a look-up service that checks if the hash code is malware. If the hashcode belongs to malware, then the latest timestamp of the malware and the rough antivirus package detection rate is returned. ArcSight SOAR uses CYMRU Malware Hash Registry Query to query computed MD5 or SHA-1 hash of a file to check for malware.

Integration Capabilities

Action

- Hash registry query

Configuration

Configuring CYMRU Malware Hash Registry Query

1. Make sure SOAR has access to CYMRU Malware Hash Registry Query integration's API as it connects to it through HTTPS.

Configuring SOAR

1. To create the integration, navigate to **Configuration** > **Integrations**.
2. Specify the following parameter values in the **Integrations Editor**:

Parameter	Value
Name	Display name of the integration
Type	CYMRU malware hash registry query
Address	Address of the integration (in the following format http[s]://malware.cymru.hash.com)

Parameter	Value
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration.
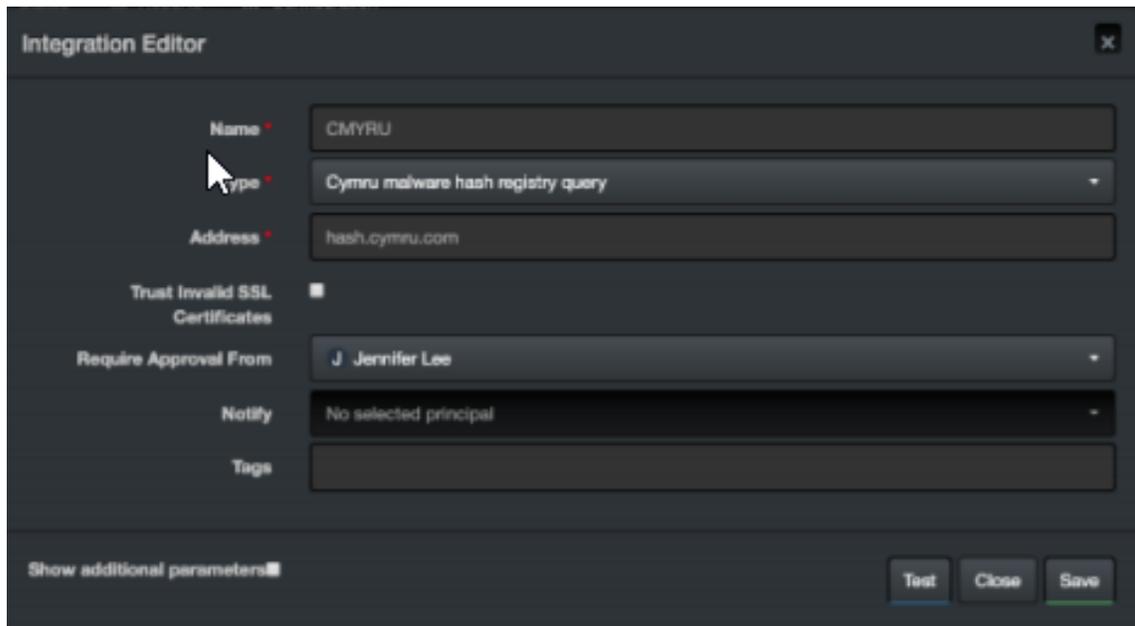
3. Click **Test** to test the integration.
4. Click **Save** to complete the integration.

Integration Guide for CyThreat Threat Intelligence

Integration Overview

CyThreat provides cyber threat intelligence data. These data feeds are enriched with subject and event-based reports as compiled by STM analysts.

CyThreat collects data from various open and commercial sources (deep/dark web, social media, blogs, forums, etc.) automatically. This allows the detection of the activities of the threat actors, proactive prevention of cyber-attacks before they occur and also allows applications to take preventive measures.

SOAR can seek benefit from CyThreat intelligence from both Integration and as Alert Source.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with CyThreat Threat Intelligence:

- Domain Query
- Hash Query
- IP Query

Alert Source Capability

ArcSight SOAR has the following alert source capability with CyThreat Threat Intelligence:

- Consume Threat Intelligence feeds from CyThreat(default)

Prerequisites

- You must have access to HTTPS as the ArcSight SOAR connects to CyThreat API through this service.
- API token and password to connect to CyThreat Threat Intelligence API.

Configuration

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**.
2. Specify the following parameters in the **Credential Editor** form:

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, CyThreat Credentials).	Empty	API password that you have received from CyThreat service.	API token that you have received from the CyThreat service.

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Specify the following parameter values in the **Configuration Form.**

Parameter	Value
Name	Display name of the CyThreart integration.
Type	CyThreat
Address	Address of the integration (the format should be https://cti.stm.com.tr).
Configuration	Specify the following configuration parameters:
	proxy.id **ID of the Proxy integration if you access https://cti.stm.com.tr through a web proxy device. For example: proxy.id = 12345** .
Credential	Credential that has been defined for this integration under the **Credentials** menu.
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers. The SSL certificate of CyThreat service is going to known by SOAR, so you do not need to check this box.
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Save** to save the integration definition.
6. Navigate to **Configuration>Customization Library** and edit **CyThreat Advanced Action Script Default Script Template**.
7. Select the integration that you have added to **Integrations** menu.
8. Click **Save** to complete the integration.

> **Note**: Steps 7-9 are required only for Advanced Action Script Default Templates.

9. Navigate to **Configuration** > **Integrations** > **CyThreat integration**.

10. Click **Test**. **Integration Successful** message is displayed if the credential and address are valid.

Configuring CyThreat as an Alert Source

1. Navigate to **Configuration** > **Alert Source** > **Create Alert Source Configuration**.

2. Select **CyThreat Threat Intelligence** and specify the following parameters in the **Alert Source Configuration Editor**:

Parameter	Value
Name	Display name of the CyThreart alert source.
Type	CyThreat Threat Intelligence
Address	https://cti.stm.com.tr/api/
Alert Severities	Arrangement table of severity mapping.
enable.ip.risk.source	Uncomment and change to true to consume IP Source.
enable.domain.risk.source	Uncomment and change to true to consume Domain Source.
enable.hash.risk.source	Uncomment and change to true to consume Hash Source.
enable.usom.blacklist.source	Uncomment and change to true to consume Usom Blacklist Source.
ip.min.risk	SOAR is not going to create case if risk level of the incoming alarm is below of the value.
domain.min.risk	SOAR is not going to create case if risk level of the incoming alarm is below of the value.
hash.min.risk	SOAR is not going to create case if risk level of the incoming alarm is below of the value.
proxy.id	ID of the Proxy integration if you access https://cti.stm.com.tr through a web proxy device. For Example: proxy.id = 12345.
days.to.look.back.at.initial.sync	How far (in days) into the past SOAR will look for remote incidents at the initial sync task.
Credential	Name of the credential set created on step 2 Configuring SOAR part (For example, CyThreat Credentials).
Visible Alert Fields	Field names from the alert if you want to show them on case.
Trust Invalid SSL Certificates	The SSL certificate of CyThreat service is going to known by SOAR, so you do not need to check this box.

3. Click **Test**. The **Alert Source tested successfully** message is displayed if your credentials are valid.

4. Click **Save**.

Integration Capabilities

1. **Domain Query**

    Enrichment capability for retrieving domain information.

    The following table presents the **Domain Query** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Domain	Domain that you want to query.	Host	Yes	No

    Output:

    Case Scope: N/A

    Human Readable Output: Yes

2. **Hash Query**

    Enrichment capability for retrieving hash information.

    The following table presents the **Hash Query** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Hash	Hash value that you want to query.	Hash	Yes	Yes

    Output:

    Case Scope: N/A

    Human Readable Output: Yes

3. **IP Query**

    Enrichment capability for retrieving domain information.

    The following table presents the **IP Query** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
IP	Ip that you want to query.	Host	Network Address	Yes

Output:

Case Scope: N/A

Human Readable Output: Yes

Integration Guide for EmailRep

Integration Overview

EmailRep consists of crawlers, scanners and enrichment services that collect data from email addresses, domains, and internet personas.

EmailRep uses hundreds of data points from social media profiles, professional networking sites, dark web credential leaks, data breaches, phishing kits, phishing emails, spam lists, open mail relays, domain age and reputation, and deliverability to predict the risk on an email address.

This integration enables ArcSight SOAR to report and query an email address.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with EmailRep:

- Email Query
- Report Email

Prerequisite

- An API key is required for accessing EmailRep.

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor** form:

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, EmailRep Credentials).			API Key

3. Click **Configuration** > **Integration** > **Create Integrations** Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of the integration.
Type	EmailRep
Address	Address of the integration (https://emailrep.io).
Configuration	Specify the following configuration parameters:
	proxy.id — **ID of the Proxy integration if you access EmailRep through a web proxy device. For example, proxy.id = 12345** .
Credential	Credential that has been defined for this integration under the **Credentials** menu.
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

4. Click **Save** to save the integration definition.

5. Navigate to **Configuration**>**Customization Library** and edit **Emailrep Advanced Action Script Default Template**.

6. Select the integration that you have added to **Integrations** menu.

7. Click **Save** to complete the integration.

8. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Email Query**
   Enrichment capability for getting reputation of email addresses.

   The following table presents the **Email Query** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Email Address	Email address to be queried.	Email Address	Yes	Yes
Do not Use Cache	SOAR does not use cached results if this box is checked.	Checkbox	N/A	No

Output:

Case Scope:

Action	Type	Category/ Value
Set	Scope item value	EmailRep Suspicious
Set	Scope item value	EmailRep Reputation

Human Readable Output:

Key	Value
Reputation	high
Suspicious	false
Domain Reputation	low
Primary MX Server	l2seng-com0i.mail.protection.outlook.com

2. Report Email

3. Action capability for reporting malicious email addresses.
   - Rollback: No
   - Duplicate Control: Yes

> **Note**: This capability requires Professional or Enterprise API membership to EmailRep.

The following table presents **Report Email** action capability details:

Output:

	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
Email Address	Email address to be reported.	Email Address	Yes	Yes
Tag	Report tag.	String	N/A	No
Description	Description/ reason to report.	String	No	Yes

Case Scope: N/A

Human Readable Output: N/A

Integration Guide for DNS Service

Integration Overview

DNS Server is used to resolve and translate the IP addresses, host names and queries to various DNS records.

Integration Capabilities

SOAR has the following integration capabilities with DNS Server.

- DNS Lookup

Configuration

Prerequisites

- Make sure SOAR has access to DNS Server through 53/udp port

Configuring DNS Service

- No specific configuration is needed on DNS Server.

Configuring SOAR

1. Click **Configuration** > **Integrations** > **Create Integrations**.
2. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of DNS Server integration on SOAR.
Type	DNS Service
Address	Address of the integration (in the format: 192.168.2.53)

Parameter	Value
Trust Invalid SSL Certificates	Not applicable
Require Approval From	Select users from the list who can provide approval before executing actions on this integration. As SOAR only executes enrichment on DNS Server, leave it empty
Notify	Select users from the list to notify when SOAR performs an action on this integration. As SOAR only executes enrichment on DNS Server, leave it empty

3. Click **Test**. The following pop up will be displayed if your credential and address are valid.
4. Click **Save** to complete integration.

Integration Guide for ESB Karar

1. To create the alert source, click **Configuration** > **Alert Source**.

2. Specify the following parameter values in the **Configuration Editor**:

Parameter	Value
Name	Display name of the alert source
Type	ESB Karar
Address	Address of the alert source. (in the format imap[s]://host:port/FOLDER).
Configuration Content	esb.imap.host= esb.imap.port= esb.imap.secure=true esb.imap.folder=INBOX esb.mail.from=info@esb.org.tr
Credential	Credential defined for this alert source under the Credentials menu
Visible Alert Field	- details.description - details.allowed

3. Click **Test**. The following pop up will be displayed if your credential and address are valid.

4. Click **Save** to complete integration.

Integration Guide for F5 Big-IP Advanced Firewall Manager

Integration Overview

Big IP AFM protects the network against incoming threats, even the most massive and complex DDoS attacks.

Big IP AFM keeps bad traffic away from some specific network addresses and protects the data center against DDoS attacks, and other network or application attacks. It also brings visibility and control to SSH, and SSL connections, providing against back door threats that use the SSH channel for data breaches and app attacks.

Integration Capabilities

Action

- Add address to specific address list

Configuration

Configuring F5 Big-IP Advanced Firewall Manager

- Make sure SOAR has access to F5 Big-IP Advanced Firewall Manager integration's API as it connects to it using HTTPS.

Configuring SOAR

1. To create the integration, navigate to **Configuration** > **Integrations**.
2. Specify the following parameter values in the **Integrations Editor** form.

Parameter	Value
Name	Display name of integration
Type	F5 Big-IP Advanced Firewall Manager

Parameter	Value
Address	Address of the integration (in the format 1.1.1.1:1234 or abc.example.com:1234)
Credential	Credential that was defined for this integration under the **Credentials** menu
Trust Invalid SSL Certificates	Select this if device's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration

3. Click **Test**. The following pop up will be displayed if your credential and address are valid.
4. Click **Save** to complete integration.

Integration Guide for FireEye HX

Integration Overview

FireEye HX is an endpoint threat detection and prevention solution. ArcSight SOAR integrates with FireEye HX through REST API to give enrichment and action capabilities to the users.

Integration Capabilities

Enrichment

- **IoC Scan:** HX can scan a given scope item in a target system and return information.
- **Detailed System Information:** HX can gather a target system information.
- **Script Execution:** HX supports different forensic data gathering scripts. These are XML formatted files that exist on HX installation. If customer wishes, they can import these script like files into Customization Library and then execute them through SOAR.

Action

Quarantine: HX quarantines a target system and reverts the quarantine if required.

Configuration

Configuring FireEye HX

- Make sure API services are enabled and create a api_admin user.To enable the service, please see product documentation
- Access to the port number defined in the HX during installation as SOAR connects to FireEye HX.
- Define required access control rules if SOAR and FireEye HX are segregated.

Configuring SOAR

SOAR configuration is standard and users need to specify **Name**, **Address** and **Credential fields**. Rest of the fields can be changed as required.

> **Note: Configuration** field must not be changed by users.

1. To create the integration, navigate to **Configuration** > **Integrations**.
2. Specify the following parameter values in the **Integration Editor** form:

Parameter	Value
Name	Display name of the integration
Type	FireEye HX
Address	Address of the alert source (in the format http[s]://1.1.1.1:3000 or http[s]://abc.example.com:3000)
Configuration	Specify the following configuration parameter: `server.address.suffix=/hx/api/v3`
Credential	Credential defined under the **Credentials menu**
Trust Invalid SSL Certificates	Select this if device's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

3. Click **Test** to test the integration.
4. Click **Save** to complete integration.

Integration Guide for Forcepoint Cloud Services

Integration Overview

SOAR works with Forcepoint Cloud Services to report uncategorized sites.

Integration Capabilities

Action

- Report

Configuration

Configuring Forcepoint Cloud Services

- Make sure SOAR has access to HTTPS as it connects to Forcepoint Cloud Services URL https://www.websense.com).

- A user account on Forcepoint/WebSense to use the **Sitelookup** tool.

Configuring SOAR

1. To create the integration, navigate to **Configuration** > **Integrations**.
2. Specify the following parameter values in the **Integrations Editor**.

Parameter	Value
Name	Display name of the integration
Type	Forcepoint Cloud Services
Address	Address of the integration (in the format http[s]://abc.example.com:3000)
Credential	Credential defined for this integration under the **Credentials** menu.

Parameter	Value
Trust Invalid SSL Certificates	Select this if device's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

3. Click **Test** to test the integration.
4. Click **Save** to complete the integration.

Integration Guide for Forcepoint Content Gateway

Integration Overview

Forcepoint Web Content Gateway is a web proxy and cache that analyzes HTTP(S) requests in real-time and passes the traffic to Filtering Service for policy enforcement.

Integration Capabilities

ArcSight SOAR has the following integration capability with Forcepoint Web Content Gateway:

- Block Access to IP Addresses, URLs and Hostnames

Use Case: Blocking Phishing Domains

SOAR checks the inbox of user's email, for phishing reports and automatically creates an incident record on the service desk. During the investigation, SOAR extracts the malicious IP addresses, domains, and URLs in the message body and blocks access to Forcepoint Web Content Gateway. This can either be performed automatically within a playbook or manually by an analyst.

Also, SOAR uses threat intelligence (TI) feeds as an Alert Source and automatically blocks malicious domains/IP addresses reported by TI source on Forcepoint Web Content Gateway before any attack occurs.

Configuration

Prerequisites

- Current version of Forcepoint Web Content Gateway.
- Access to HTTPS as SOAR connects to Forcepoint Web Content Gateway Policy API
- Access to 15873/tcp port

Configuring Facepoint Web Content Gateway

1. Forcepoint Management API does not get installed by default. To complete the integration, install this service on the server or appliance. Also, the configuration steps differ with the usage of the server. For the complete instructions, see Management API Installation Guide.

2. After installing Management API components, use the Forcepoint Security Manager to configure the account used for authentication. To enable the communication, see **Enabling communication between Management API clients and servers** in the Management API Installation Guide.

Configuring SOAR

1. Click **Configurtion** > **Credentials** > **Create Credential**.

2. Specify the following parameter values in the **Credential Editor** form:

    a. **Internal credential:**

Parameter	Value
Type	Internal credential
Name	Display name of the credential set (For example, Forcepoint WCG Credentials)
Username	Username configured on Forcepoint Management API
Password	Password for the user configured on Forcepoint Management API.
Private Key	Empty

    b. **Credential Store:**

Parameter	Value
Type	External credential
Name	Name of the credential with pull path of the safe on store.

3. Click **Configuration** > **integrations** > **Create Integration.**

4. Specify the following configuration parameter values in the **Configuration form**:

Parameter	Value
Name	Display name of Forcepoint Web Content Gateway integration on SOAR
Type	Forcepoint Web Content Gateway
Address	Address of the integration (in the format https://192.168.2.99:15:15873).

Parameter	Value
Configuration	Specify the following configuration parameters: ``` # The Category name cannot include any of the following characters: # * < > { } ~ ! $ % & @ # . "	\ & + = ? / ; : , # SOAR is going to automatically add new category name if it doesn't exist categoryName=SOAR_BLOCK ```
Credential	Name of the credential set created on step 2. (For example, Forcepoint WCG Credentials)
Trust Invalid SSL Cerificates	Select this if Engine's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select users from the list who can provide approval before executing actions on this integration.
Notify	Select users from the list to notify when SOAR performs an action on this integration

5. Click **Test**. The following pop up will be displayed if your credentials and address are valid.

6. Click **Save** to complete integration.

Additional Notes

- The **categoryName** you provide in the Configuration section is API-Managed but not managed by UI. If the category does not exist on the device, SOAR creates it automatically.

Integration Guide for ForeScout CounterACT NAC

Integration Overview

ForeScout CounterACT NAC provides virtual insight into any device connected across the enterprise and gives a single-pane-of-glass perspective. ForeScout discovers devices in real-time, then classifies, assesses, and monitors these devices. Also, this platform provides agent-less control and continuous monitoring across heterogeneous environments. Enables to trigger actions to notify, monitor, and remediation.

Integration Capabilities

SOAR has the following integration capability with ForeScout CounterACT NAC:

Action Capabilities

- Assign Policy to Host

Enrichment Capabilities

- Host information query by Network Address

- Host information query by Username

- Host information query by MAC Address

- Host information query by Computer Name

Use Case: Isolating Mal-behaving PC

SOAR integrates with ForeScout CounterACT NAC, while responding to an incident it applies a policy to mal-behaving computers and prevents further spread of the attack. A policy to the host can either be applied automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Current version of ForeScout CounterACT NAC

- Access to SSH protocol(22/tcp port) as SOAR connects to ForeScout CounterACT NAC using SSH protocol.

- Access to 443/tcp port as enrichment plugin connects to ForeScout CounterACT NAC server

- A shell user account needs to be created for SOAR to connect to ForeScout

CounterACT NAC

Configuring ForeScout CounterACT NAC

1. Login to ForeScout CounterACT NAC appliance.
2. Create a shell account by executing the following command in the command prompt:

   ```
   $ useradd -s /bin/bash -m -d /home/soar soar
   ```

   ```
   $ passwd atar
   ```

3. To allow new user to execute fstool command without the need to enter the password, add the following line to sudo configuration (/etc/sudoers)

   ```
   soar ALL=(root) NOPASSWD: /usr/local/forescout/bin/fstool
   ```

4. To use enrichment capabilities, add or use an existing web management user with the following permission:
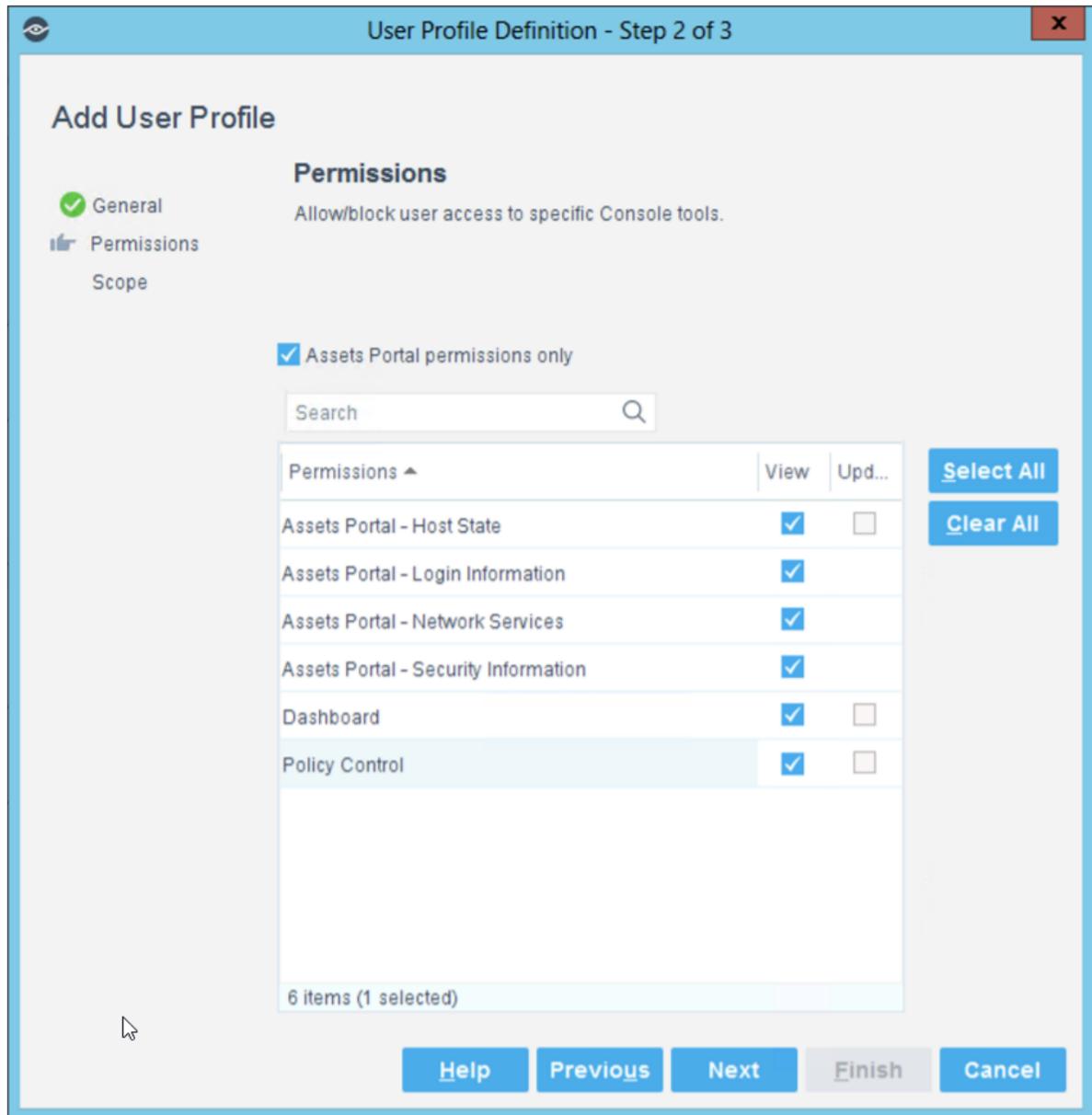
5. Login to Forescout **Management Interface.**
6. Enable **CEF service**.
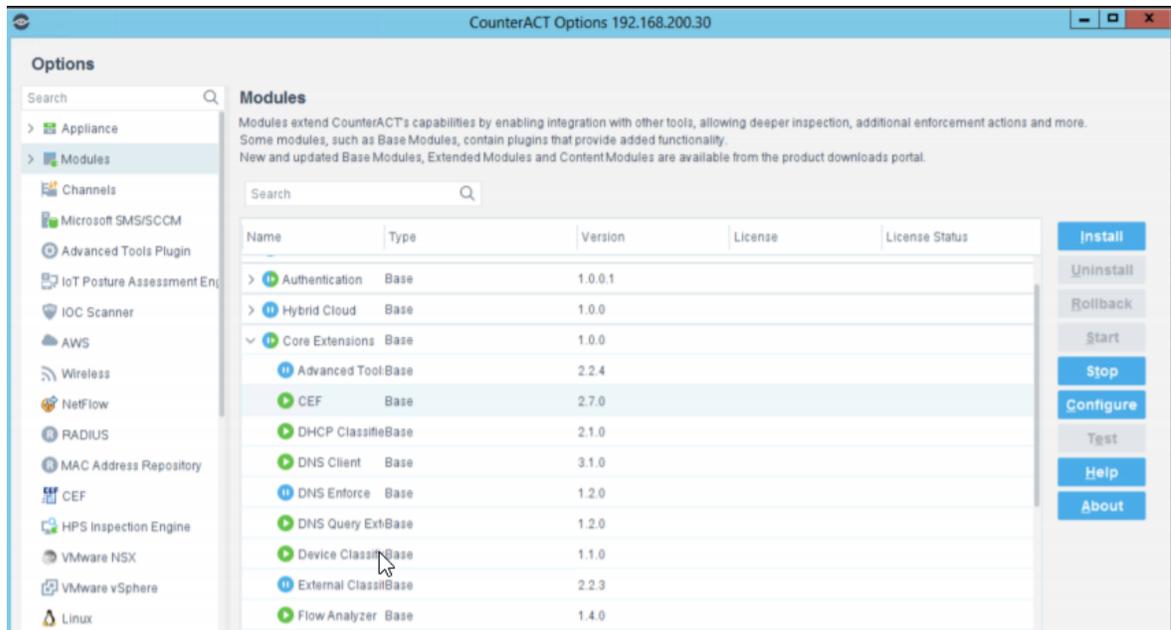
7. Navigate to **Policy** and edit one of the existing policies or create a new one.

8. To edit condition of a rule, add "SIEM Message" as Criteria and select desired action.

> **Note:** Make a note or save the SIEM message to use while configuring SOAR.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential.**

2. Specify the following parameter values in the **Credential Editor** form:

   a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, ForeScout CounterACT NAC Credential)
Username	Username created for SOAR on ForeScout CounterACT NAC
Password	Password of the user that was created for SOAR on ForeScout CounterACT NAC
Private Key	Empty

   b. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, ForeScout CounterACT NAC Credential)
Username	Username created for SOAR on ForeScout CounterACT NAC for web management user (2.2.3).
Password	Password of the user you have created for SOAR on ForeScout for web management user (2.2.3).
Private Key	Empty

> **Note:** Make a note or save the credential ID to use it in device configuration (2.3.4).

c. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store.

3. Click **Confiugration** > **Integrations** > **Create Integration.**

Specify the following parameter values in the **Configuraiton** form:

Parameter	Value
Name	Display name of Database Server integration on SOAR
Type	ForeScout CounterACT NAC
Address	Address of the integration (in the format 192.168.1.1)

Configuration	Specify the following configuration parameters.
	
	``# Supported versions are: v1 (for version 8.0) and v2 (for version 8.1.3).
	Default version is v1
	#version=
	# Siem messages should be separate with comma
	# For Example:
	# policy.siem.messages=MSG1,MSG2,MSG3
	policy.siem.messages=
	# please provide the credential id if the ForeScout query page has a
	# different username & password
	webui_credential_id=(Credential id that you made a note in step 2.3.4)```

Credential	Name of the credential set created on step 2. (For example, ForeScout CounterACT NAC Credential)

Parameter	Value
Trust Invalid SSL Certificates	Select this if device's certificate is self-signed or is not recognized by browsers
Require Approval from	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration.

4. Click **Save** to complete integration.

5. Click **Test** to test the integration.

Additional Notes

- ForeScout CounterACT NAC integration is an Advanced Script, and the content of the default script is accessible under **Configuration** > **Customization** Library.

- While defining the integration for the first time, you might encounter the following warning message, which is the expected behavior for this type of integration.
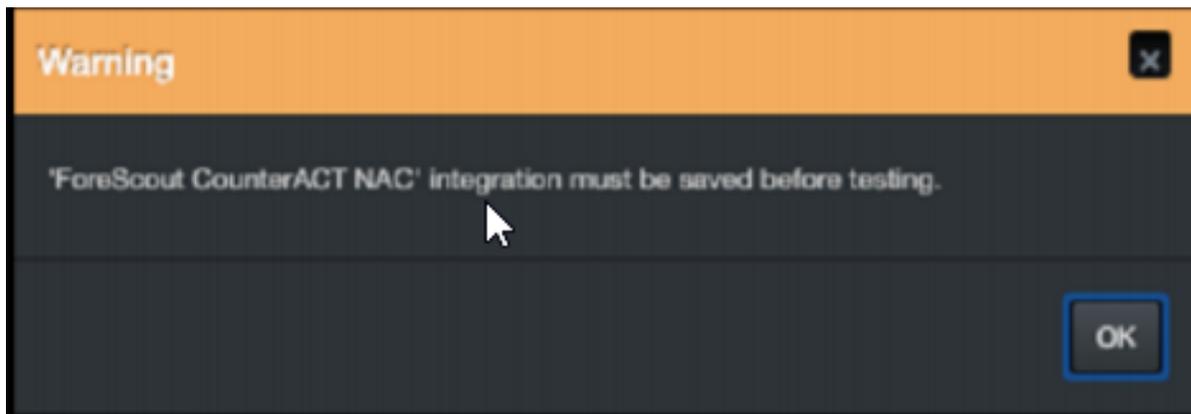
Integration Guide for Fortinet Forti Manager V2

Integration Overview

FortiManager is a management tool for Fortify Firewalls. It can manage multiple firewalls in a row from its central user interface.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Forti Manager:

- Add to Address Group
- List Devices
- List Firewall Address Groups
- List Firewall Addresses

Prerequisites

- You must have access to HTTPS as the ArcSight SOAR connects to Forti Manager IP through this service.
- You must have a super user credentials.

Configuration

Configuring Forti Manager

SSH to FortigateManager with admin user credential and execute the following command on ssh terminal:

```
FW # config system admin user
(user)# edit admin
(admin)# set rpc-permit read-write
```

Configuring SOAR

1. Click **Configuration** > **Integration** > **Create Integration**.

2. In **Configuration Editor**, select **FortiManager** in the **Type** list.

3. Click **Create** to create a new credential and specify the following parameters in the **Credential Editor** form:

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, Fortin Manager Credentials).	FortiManager Username	FortiManager Password	Empty

4. Check the **Clear Text Access**checkbox .

5. Click **Save** to save the integration definition.

6. Navigate to **Configuration**>**Customization Library** and edit **FortiManager Advanced Action Script Default Template**.

7. Select the integration that you have added to **Integrations** menu.

8. Click **Save** to complete the integration.

9. Click **Test**. **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Add To Address Group**

2. Adds Ip address to given group for specified ADOM.
   The following table presents the **Add To Address Group** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
IP	A valid IP Address to retrieve data.	Network Address Host	Yes	Yes
ADOM	Administrative Domain.	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

3. List Firewall Address Groups

4. List of firewall address groups on FortiManager.
   The following table presents the **List Firewall Address Groups** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
ADOM	Administrative Domain.	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output:

5. List Firewall Addresses

6. List of Firewall Addresses on FortiManager.
   The following table presents the **List Firewall Addresses** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
ADOM	Administrative Domain.	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output:
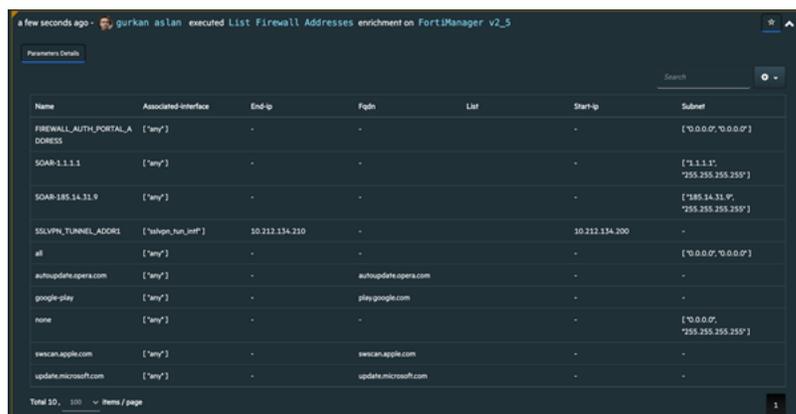
Integration Guide for Fortinet FortiGate Firewall

Integration Overview

ArcSight SOAR uses Fortinet FortiGate Firewall to block IP addresses on the network perimeter and terminates sessions using the incident scope.

Integration Capabilities

- Action
- Block
- Disconnect
- Custom Script

Configuration

Configuring FortiGate Firewall

- Make sure SOAR has access to SSH as it connects to FortiGate Firewall integration using it
- A user's credential with admin role
- An empty rule to be used by SOAR to block offending IP addresses

Configuring SOAR

1. To create the integration, navigate to **Configuration** > **Integrations**.
2. Specify the following parameter values in the **Integrations editor**:

Parameter	Value
Name	Display name of the integration
Type	Fortigate Firewall
Address	Address of the integration (in the following format: 1.1.1.1 or abc.example.com)
Credential	Credential defined under the **Credentials menu**

Parameter	Value
Trust Invalid SSL Certificates	Select this if device's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

3. Click **Test** to test the integration.

4. Click **Save** to complete the integration.

Additional Notes

You might have to review the actions that are defined and executed using the Fortigate Firewall custom scripts for SOAR. To access these custom scripts, navigate to **Configuration** > **Custom Scripts**.

The following custom scripts are specific to this device:

- FortiGate Firewall SSH Device Action (Block) Default Template
- FortiGate Firewall Availability Check Default Template

Integration Guide for Fortinet FortiAnalyzer

Integration Overview

Fortinet FortiAnalyzer is a central log collection and analysis tool for Fortinet products. SOAR can query FortiAnalyzer (FAZ) for scope items to enrich incident data and to search the past events for emerging threats.

Integration Capabilities

ArcSight SOAR has the following enrichment capabilities with Fortinet FortiAnalyzer:

- Accepted Traffic Logs : This query returns accepted traffic logs to or from the selected scope item and the time frame might be between 1 hour to 12 hours.
- URL Access Logs : This query returns the events that record access to the selected URL and the time frame might be between 1 hour to 12 hours.

Configuring Fortinet FortiAnalyzer

Web services must be enabled on the network interface to which the client connects.

1. To enable web services for an interface, navigate to **System Settings** > **Network** > **Interface**.
2. Select **Edit** for the interface for which you need to enable the web services.
3. In the **Administrative Access** section, select **Web Service**.
4. Select **OK** to apply the changes.
5. Create a user with a custom profile.

> **Note:** This user profile requires access to **Log View/FortiView/NOC - SOC** component and **ADOM's** SOAR.

Configuring SOAR

1. Click **Configurtion** > **Credentials** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor** form:

a. **Internal credential:**

Parameter	Value
Type	Internal credential
Name	Display name of the credential set (For example, Fortinet FortiAnalyzer)
Username	API Key created on Fortinet FortiAnalyzer
Password	API Password for the key created on Fortinet FortiAnalyzer
Private Key	Empty

b. **Credential Store:**

Parameter	Value
Type	External credential
Name	Name of the credential with pull path of the safe on store

3. Click **Configuration** > **integrations** > **Create Integration**.

4. Specify the following configuration parameter values in the **Configuration form**:

Parameter	Value
Name	Display name of Fortinet FortiAnalyzer integration on SOAR
Type	Fortinet FortiAnalyzer
Address	Address of the integration (in the following format: 1.1.1.1 or http[s]://abc.example.com)
Credential	Name of the credential set created on step 2 (for example, Fortinet FortiAnalyzer Credentials)
Configuration	Specify the following configuration parameters: maxNumMatches: Define the number of results SOAR shows per page of query adom: ADOM's SOAR query to get logs from
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

5. Click **Test** to test the integration.
6. Click **Save** to save the integration.

Integration Guide for Fortinet FortiDDoS

Integration Overview

FortiDDoS is a network behavior anomaly (NBA) prevention system that detects and blocks attacks that intend to disrupt network service (distributed denial of service (DDoS) attacks) by over utilizing server resources.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with FortiDDoS:

- Block IP and Hostname

Use Case: Blocking malicious IP on peripheral

SOAR integrates with FortiDDoS to block malicious IP addresses detected while responding tp an incident. Blocking can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- FortiDDoS version 4.7 and 5.1
- Access to tcp port 443 as SOAR connects to FortiDDoS' API using HTTPS
- An administrator user account for SOAR to connect to FortiDDoS

Configuring FortiDDoS

1. To add a new SOAR user with the required access profile permissions, navigate to **System** > **Admin** > **Access Profile**.

2. In the Access profile form, select **Global Settings** and **Protection profiles**with **Read & Write** permissions.

3. Navigate to **System** > **Admin** > **Administrator**.

4. To add an administrator with the profile created in the previous step, select **Enable** for **Allow API Access**.

5. (Optional) To specify **Remote Authentication** and **Idle timeout** values, navigate to **Centralized Management** > **Admin**.

6. Click **Save** to save the changes.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.

2. Specify the **Credential Editor** with the following parameter values:

   a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, FortiDDoS Credentials)
Username	User created on FortiMail for SOAR
Password	Password of the user that was created for SOAR on FortiMail
Private Key	Empty

   b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of FortiDDoS integration on SOAR
Type	FortiDDoS
Address	Address of the integration (in the following format: https://192.168.3.99)

Configuration	Specify the following configuration parameters:
	
	`# Supported API versions are: v1 (for 4.x versions)`
	`and v2 (for 5.x versions). Default api.version=v2`
	`#proxy.id=123`

Credential	Name of the credential set created on step 2 (For example, FortiDDoS Credentials)
Trust Invalid SSL Certificates	Select this if Integrations's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

5. Click **Test** to test the integration.

6. Click **Save** to complete the integration.

Integration Guide for Fortinet FortiGate API

Integration Overview

Fortinet FortiGate is an industry leading next generation security platform.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Fortinet FortiGate API:

- Action
- Block IP
- Block FQDN
- Block URL

Use Case: Blocking malicious artifacts detected through alerts

SOAR automatically executes playbooks and blocks malicious artifacts on FortiGate platform. The artifacts IP, Domain and URL can be blocked using SOAR.

Configuration

Prerequisites

- Access to tcp port 443 as SOAR connects to Fortinet FortiGate API using HTTPS
- A user account with necessary permissions on the FortiGate platform

Configuring Fortinet FortiGate

1. To create a user, navigate to **System** > **Administrators**.
2. Click **Create New** and select **REST API Admin**.
3. Specify the following values in the **New REST API Admin** form:

   Username: <SOAR user name>

   Administrator Profile: <profile name>

   Trusted Hosts: A subnet that covers SOAR's API address

> **Note:** Use the IP address that SOAR uses and **0.0.0.0/0** must not be used as an IP address.

4. To create a profile, click **+** in the **Admin Profile** window.

5. Select **Read/Write** persmissions for the following groups:

   Firewall > **Address**

   Security > **Web Filter**

6. Click **OK** to save the profile and save the API key.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.

2. Specify the **Credential Editor** with the following parameter values:

    a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Fortinet FortiGate Credentials)
Username	Empty
Password	Empty
Private Key	Enter the API Key generated by FortiGate

> **Note:**Fortinet FortiGate requires private key and External Credential is not used.

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Fortinet FortiGate integration on SOAR
Type	Fortinet FortiGate 6.0
Address	Address of the firewall
Configuration	Specify the following configuration parameters: `group.name:` Group name for adding objects to be blocked. This Address Group will be created on FortiGate and then can be used in policies as the admin see fit `policy.names:` Policy names to be used to block URL. '|' is used as separator for policies and SOAR writes the URL to all the policies defined
Credential	Name of the credential set that was created on step 2 (For example, Fortinet FortiGate Credentials)
Trust Invalid SSL Certificates	Select this if Integrations's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration

5. Click **Save** to complete the integration.

Additional Notes

- The API Key to work properly requires access to HTTPS and for security reasons as well.

> **Note:** By default, HTTP access is enabled in FortiGate. However, in production environment, it is recommneded to disable the HTTP access.

- If you have multiple policies on the integration configuration and if one of the policy's URL filter is disabled, SOAR with Fortinet integration displays no specific error message. In such case, you might encounter the following error message:

  None of policy names in t he configuration are present in the Fortinet FortiGate server.

Integration Guide for Fortinet FortiMail

Integration Overview

Fortinet FortiMail secure email gateway utilizes the latest technologies and security services from FortiGuard Labs to protect from common and advanced threats while integrating robust data protection capabilities to avoid data loss.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with FortiMail:

- Add to Block List
- Block

Use Case: Blocking malicious sender

SOAR integrates with FortiMail to block malicious email addresses detected while responding to an incident. The blocking can either be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- FortiMail version 6.2.2(GA) and later
- Access to tcp port 443 as SOAR connects to FortiMail API using it
- An administrator user account for SOAR to connect to FortiMail

Configuring FortiMail

1. By default, REST-API service is disabled on FortiMail. To enable it, use the following CLI command:

```
config system global
   set rest-api enable
   end
```

2. Navigate to **System** > **Administrator** > **Admin Profile**.

3. Select **Policy**, **Block/Safe List** with **Read-Write** support and create an admin profile in the **Admin Profile** form.

4. Navigate to **System** > **Administrator** > **Administrator**.

5. Create a new administrator account with the profile that you have created in the previous step.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.
2. Specify the **Credential Editor** with the following parameter values:

   a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, FortiMail Credentials)
Username	User created on FortiMail for SOAR
Password	Password of the user created on FortiMail for SOAR
Private Key	Empty

   b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of FortiMail integration on SOAR
Type	FortiMail
Address	Address of the integration (in the following format: https://192.168.3.100)
Configuration	Specify the following configuration parameters: #proxy.id=5433
Credential	Name of the credential set created on step 2 (For example, FortiMail Credentials)
Trust Invalid SSL Certificates	Select this if Integrations's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

5. Click **Test** to test the integration.

6. Click **Save** to complete the integration.

Additional Notes

Add to Block List capability uses the **Security** > **System** > **Blocklist**, whereas **Block capability** uses the **Policy** > **Access Control**.

Integration Guide for Fortinet FortiManager

Integration Overview

Fortinet FortiManager is a centralized management unit for Fortinet family devices. It provides best compliance practices and workflow automation.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with FortiManager:

- Block file on an connected Fortinet family device (For example, Fortinet NGFW, Fortinet FortiMail, etc)
- Block IP address on an connected Fortinet family device (For example, Fortinet NGFW, Fortinet FortiMail, etc)
- Block username on an connected Fortinet family device (For example, Fortinet NGFW)
- Block email on an connected Fortinet family device (For example, Fortinet FortiMail)

Use case: Mitigating Compromised Account Cases

SIEM, with the help of intelligence sources, creates an alarm. It compromises the suspected email accounts of the employees. SOAR integrates with Fortinet FortiManager and automatically blocks the outgoing emails and the incoming and outgoing traffic. This blocking can either be performed automatically within a playbook or manually by an analyst.

Prerequisites

- Fortinet FortiManager v5.6.2-build1631 180124 (GA) firmware version as SOAR supports it
- Access to tcp port 443 as SOAR connects to Fortinet FortiManager using HTTPS
- A user account for SOAR to connect to Forti Manager

Configuration

Configuring FortiManager

1. Navigate to **System Settings** > **Admin** > **Administrators**.
2. To create a profile with Super_User account, specify the following values in the **New Administrator** form:

   - **Username:** <SOAR username>

   - **Admin Type:** Local

   - **New Password:** <Specify the password>

   - **Confirm Password:**< Confirm the password entered in the **Password** field>

   - **Admin Profile:** Super_User

3. Navigate to **System Settings** > **Network**.
4. Enable the **Web Service** in the **Administrative Access**.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.

2. Specify the following parameter values in the **Credential Editor**:

   a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Forti Manager Credentials)
Username	User that was created for SOAR on Forti Manager
Password	Password of the user that was created for SOAR on Forti Manager
Private Key	Empty

   b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of FortiMail integration on SOAR
Type	Forti Manager
Address	Address of the integration (in the following format: https://192.168.2.2:8080)

Parameter	Value
Credential	Name of the credential set created on step 2 (For example, Forti Manager Credentials)
Trust Invalid SSL Certificates	Select this if Forti Manager's certificate is self signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

5. Click **Test** to test the integration.
6. Click **Save** to complete the integration.

Additional Notes

Commands to be run on Forti Gate firewall devices are defined as Advanced Action Script. To access the default scripts navigate to **Configuration** > **Customization Library**.

Integration Guide for Fortinet FortiSandbox

Integration Overview

Fortinet Sandbox is a zero-day malware behavior analysis system. It enables organizations to defend against advanced threats such as ransomware by integrating various Fortinet technologies and other security products. Or is used as an extension to their on-premise security architectures to leverage complete control.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Fortinet Sandbox:

- Query File Hash
- Analyze File
- Analyze URL

Use Case: Investigating Suspicious Files

During the investigation of a suspicious endpoint behavior, SOAR integrated with Fortinet Sandbox analyzes the behavior of potential malware and hashes and URLs detected on suspicious network traffic. This investigation can either be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Fortinet Sandbox 3.1.0 version as SOAR supports it
- Access to tcp port 443 as SOAR connects to Fortinet Sandbox API using HTTPS
- A user account is required for SOAR to connect to Fortinet Sandbox

Configuring Fortinet Sandbox

1. Navigate to **System** > **Admin Profiles**.
2. Create an Admin Profile with **Read/Write permission** for **SCAN INPUT** and select **Enable** for **JSON API**.
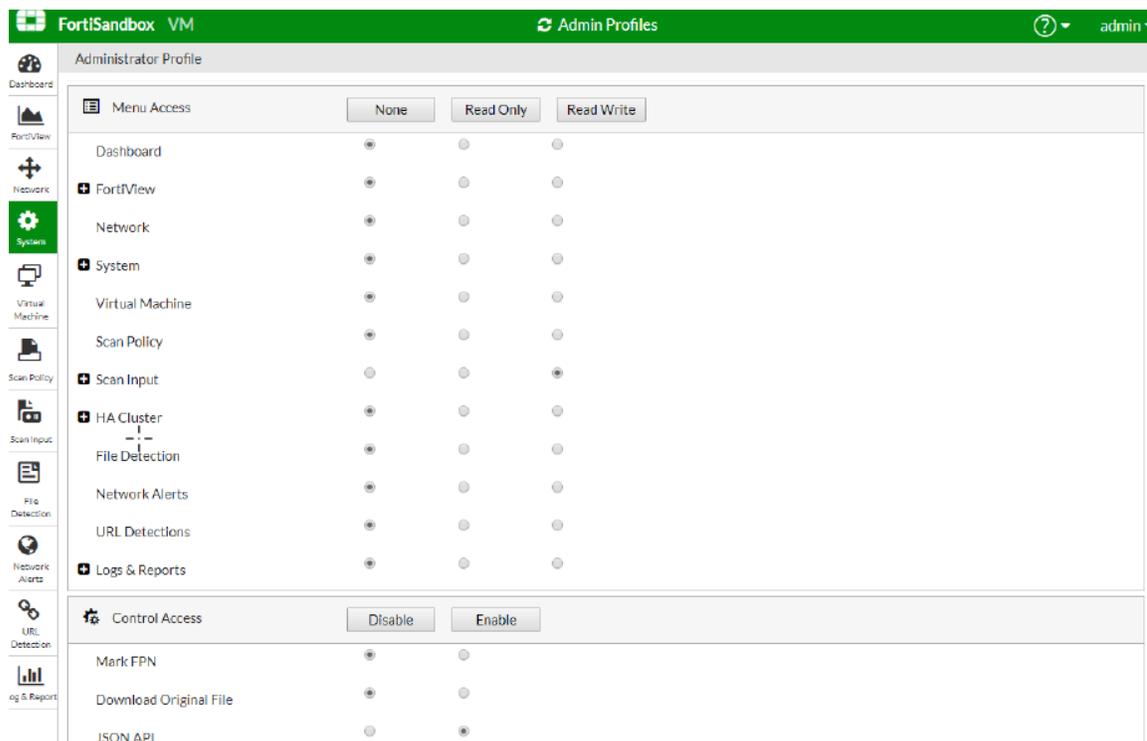
3. Navigate to **System** > **Administrators**.

4. Create an **Administrator** account with the profile that is created in the previous step and specify the following values:

   - **Administrator:** SOAR_LABS

   - **Password:** <Specify the password>

   - **Confirm Password:** <Confirm the password spcified in the Password field>

   - **Type:** Select **Local**

   - **Admin Profile:** <Specify the profile name>
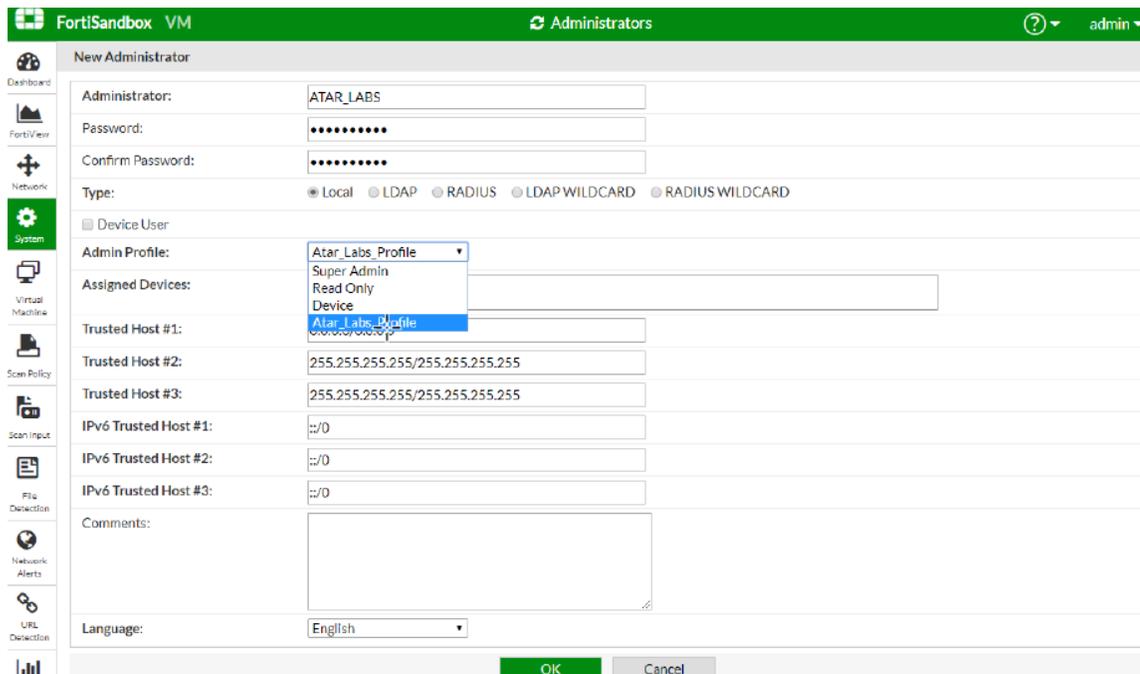
Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.

2. Specify the following parameter values in the **Credential Editor**:

    a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Fortinet Sandbox Credentials)
Username	User that was created on Fortinet Sandbox for SOAR
Password	Password of the user that was created for SOAR on Fortinet Sandbox
Private Key	Empty

    b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Fortinet Sandbox integration on SOAR
Type	Fortinet Sandbox
Address	Address of the integration (in the following format: https://192.168.2.75)
Configuration	Specify the following configuration parameters: #proxy.id=5442
Credential	Name of the credential set created on step 2 (For example, Fortinet Sandbox Credentials)
Trust Invalid SSL Certificates	Select this if Web UI's certificate certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Not Applicable
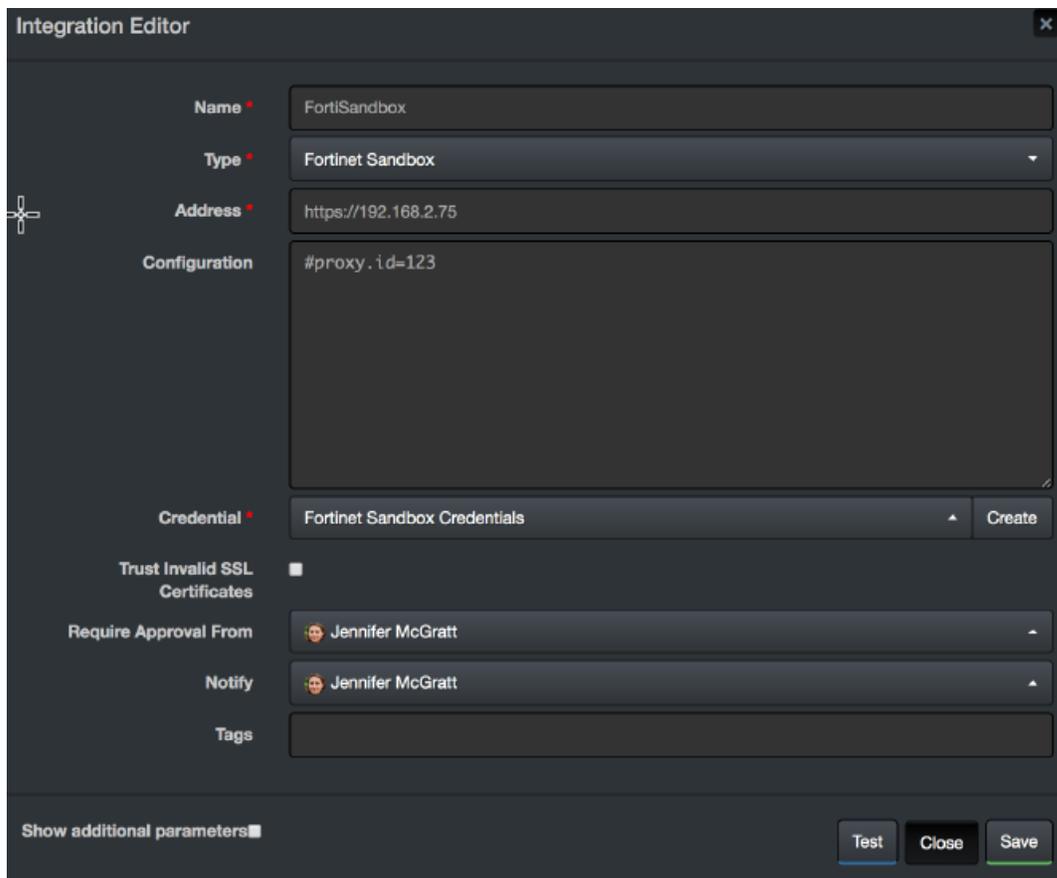
5. Click **Test** to test the integration.
6. Click **Save** to complete the integration.

Additional Notes

Fortinet Sandbox supports the following compressed file types:

.tar, .z, .xz, .gz, .tar.gz, .tgz, .zip, .bz2, .tar.bz2, .tar.Z, .7z, .rar, .lzh, .ace

Integration Guide for FTP Server

Integration Overview

ArcSight SOAR uses FTP Servers to put or transfer files to remote machines using incident scope.

Integration Capabilities

Action

- Put File

Configuration

Prerequisites

- Access to File Transfer Protocol or SFTP as SOAR connects to FTP Server using it
- A user's credential

Configuring SOAR

1. To create the integration, navigate to **Configuration** > **Integrations**.
2. Specify the following parameter values in the **Integrations Editor** form.

Parameter	Value
Name	Display name of the integration
Type	FTP Server
Address	Address of the integration (in the format: 1.1.1.1 or abc.example.com)

Parameter	Value

Configuration	Specify the following configuration parameters:
	
	`connection.port is the listening port of the FTP/SFTP service running.`
	`connection.protocol could be FTP or SFTP.`
	`remote.file.filename.appenduuid specifies whether the UUID will be appended to the filename. It can be either "true" or "false".`
	`remote.folder is the folder relative to the FTP home directory.`

Credential	Credential that was defined for this integration under the **Credentials** menu
Trust Invalid SSL Certificates	Select this if device's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

3. Click **Test** to test the integration.

4. Click **Save** to complete integration.

Integration Guide for Have I Been Pwned

Integration Overview

Have I Been Pwned is a web service that allows to check if the emails/usernames are exposed as part of previous data breaches.

This integration supports Have I Been Pwned API v3.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Have I Been Pwned:

- Check Pwned Accounts
- Check Pwned Pastes
- Check Pwned Domains

Prerequisites

Have I Been Pwned requires an API key for access.

Configuration

Configuring SOAR

1. **Click Configuration** > **Credential** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor** form:

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (i.e., Have I Been Pwned Credentials)			API Key

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration Form**:

Parameter	Value
Name	Display name of the integration
Type	Have I Been Pwned
Address	Address of the integration (Have I Been Pwned)
Configuration	Specify the following configuration parameters:
	proxy.id Access the ID of the Proxy integration Have I Been Pwned through a web proxy device. For example: proxy.id = 12345
Credential	Credential that has been defined for this integration under the Credentials menu.
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers.
Require Approval From	Since there is no action capability in this plugin, please leave it empty.
Notify	Since there is no action capability in this plugin, please leave it empty.

5. Click **Save** to save the integration definition.

6. Navigate to **Configuration>Customization Library** and edit **Have I Been Pwned Advanced Action Script Default Template**.

7. .Select the integration you have added to **Integrations** menu.

8. Click **Save** to complete the integration.

9. Click **Test**. **Integration Successful** message is displayed if your credential and address are valid.

Capabilities

1. **Check Pwned Accounts**
   Enrichment capability for gathering pwned account details.

   The following table presents the **Check Pwned Accounts** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration	Integration	N/A	Yes
Email Address	Email address to be queried	Email Address Username Keyword Unknown	Yes	Yes
Do not Use Cache	SOAR does not use cached results if this box is checked	Checkbox	N/A	No

Output:

Case Scope: N/A

Human Readable Output:

2. **Check Pwned Domains**

   Enrichment capability for gathering pwned domain details.

   Following is the **Check Pwned Domains** enrichment capability details.

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration	Integration	N/A	Yes
Domain	Domain to be queried	Domain Keyword Unknown	Yes	Yes
Do not Use Cache	SOAR does not use cached results if this box is checked	Checkbox	N/A	No

Output:

Case Scope: N/A

Human Readable Output:

3. **Check Pwned Pastes**

   Enrichment capability for listing the paste sites that pwned account is mentioned.

   Following is the **Check Pwned Pastes** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration	Integration	N/A	Yes
Email Address	Email address to be queried	Email Address Username Keyword Unknown	Yes	Yes
Do not Use Cache	SOAR does not use cached results if this box is checked	Checkbox	N/A	No

Output:

Human Readable Output:

Integration Guide for Generic HTTP SMS Gateway

Integration Overview

ArcSight SOAR uses Generic HTTP SMS (Short Message Service) Gateway to send SMS.

Integration Capabilities

- None

Configuration

Configuring Generic HTTP SMS Gateway

- Access to **File HTTPS** service as SOAR uses it to connect to Generic HTTP SMS Gateway
- A SOAR user account

Configuring SOAR

1. To create the integration, navigate to **Configuration** > **Integrations**.
2. Specify the following parameter values in the **Integrations Editor** form.

Parameter	Value
Name	Display name of the integration
Type	Generic HTTP SMS Gateway
Address	Address of the integration (in the following format: 1.1.1.1 or abc.example.com)
Configuration	Specify the following configuration parameters:

```
http.method = POST
http.auth.enabled = false
params.jobID = ${credential.privateKey}
params.url = http://dev.swh.soarlabs.io/atar/
params.username = ${credential.username}
params.text = ${text}
params.gsmNumber = ${recipient}
http.header.User-Agent = SOAR
http.header.Content-Type = application/x-www-form-urlencode
sms.stripCountryCode = +90
```

Parameter	Value
Credential	Credential that was defined for this integration under the **Credentials** menu
Trust Invalid SSL Certificates	Select this if device's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

3. Click **Test** to test the integration.
4. Click **Save** to complete integration.

Integration Guide for HTTP Proxy

Integration Overview

ArcSight SOAR uses HTTP proxies to access HTTP services. Some integration plugins are capable of accessing resources on the Internet or other networks through a proxy device configuration. See the respective integration guides for configuring the proxy.

Configuration

Prerequisites

- Access to proxy service for SOAR
- A user account to connect to proxy if proxy authentication enabled

Configuring HTTP Proxy

HTTP Proxy software must be configured to get the access to SOAR. You can consult the system to know the HTTP Proxy used in the network.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor**:

   Internal Credential

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (for example, HTTP Proxy Credentials)
Username	User that was created on HTTP proxy software for SOAR
Password	Password of the user that was created on HTTP proxy software for SOAR
Private Key	Empty

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of HTTP Proxy integration on SOAR
Type	HTTP Proxy
Address	Address of the integration (in the following format: https://192.168.1.3:8081)
Configuration	Specify the following configuration parameters: ```# Supported values: basic, ntlm, none\n# For NTLM, username in credential should be specified\nlike: username@domain\nauthentication.type=basic\n# URL to use when testing availability of this proxy\nintegration.\n# Defaults to the value of HttpProxyCheckURL\nconfiguration parameter.```
Credential	Name of the credential set created on step 2 (For example, HTTP Proxy Credentials)
Trust Invalid SSL Certificates	Select this if Web UI's certificate certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

5. Click **Test** to test the integration.
6. Click **Save** to complete the integration.

Additional Notes

For SOAR to perform Automatic Update Checks, navigate to **Configuration** > **Parameters** and set ProxyIntegrationIdForAutomaticUpdateCheck.

Integration Guide for IBM Security X-Force

Integration Overview

IBM X-Force Exchange is a cloud-based threat intelligence platform that enables users to research security threats, search attack indicators, aggregate actionable intelligence, and collaborate with peers.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with IBM X-Force Exchange:

- DNS Records
- IP Report
- Malware for File Hash
- Send File for Analysis
- URL Report

Use Case: Investigating Phishing Campaigns

SOAR follows the user's email inbox for phishing reports and automatically creates an incident record on its service desk. While investigating the attack, SOAR extracts the sender address, IP address, URLs in the message body, files in the attachment, and checks with IBM X-Force Exchange if these attacks are previously analyzed. This investigation can either be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Access to https://api.xforce.ibmcloud.com (443/tcp port) for SOAR to connect to IBM X-Force Exchange API
- An API key for SOAR to connect to IBM X-Force Exchange

Configuring IBM X-Force Exchange

1. Log in to https://exchange.xforce.ibmcloud.com.

2. To create a new API key, navigate to **Settings** > **API Access**.

> Note: Save the generated API key and the password.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.

2. Specify the following parameter values in the **Credential Editor**:

   a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, IBM X-Force Exchange Credentials)
Username	API Key created on IBM X-Force Exchange
Password	API Password for the key created on IBM X-Force Exchange
Private Key	Empty

   b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of IBM X-Force Exchange integration on SOAR
Type	IBM X-Force Exchange
Address	Address of the integration (https://api.xforce.ibmcloud.com)
Configuration	Specify the following configuration parameters: ``` # Integration ID of the proxy integration to use when connecting # to current integration. # If not provided, SOAR will try to use a direct connection. #proxy.id=123 # configure how far (in minutes) into the past this enrichment will look. cache.reusing.duration=60 ```
Credential	Name of the credential set created on step 2 (For example, IBM XForce Exchange Credentials)
Trust Invalid SSL Certificates	Select this if Web UI's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

5. Click **Test** to test the integration.

6. Click **Save** to complete the integration.

Integration Guide for Infoblox DNS Firewall

Integration Overview

Infoblox DNS Firewall defends DNS servers from the comprehensive range of DNS-based attacks while maintaining service availability and business continuity. The Grid Manager web interface provides access to the appliance for network and IP address management.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Infoblox DNS Firewall:

- Block IP address (No Data)
- Block IP Address (No Such Domain)
- Block Host (No Data)
- Block Host (No Such Domain)
- Substitute DNS A Record

Use Case: Blocking malicious IP addresses on DNS

SOAR integrates with Infoblox DNS firewall to block malicious IP addresses and hosts on DNS firewall to stop malware attacks and protect users. These actions can either be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Infoblox NIOS 8.4 version
- Access to tcp port 443 as SOAR connects to Infoblox DNS Firewall API
- A SOAR user account to connect Infoblox DNS Firewall

Configuring Infoblox DNS Firewall

1. Navigate to **Administration** > **Administrators** > **Admins**.
2. To add an account, specify the following values in the **Add Administrator Wizard**:

   Authentication Type: Local

Login: <Specify the username>

Password: <Specify the password>

Confirm Password: <confirm the password specified in **Password** field>

Admin Group: Select *admin-group*

3. To create a new Response Policy Zone, navigate to **Data Management** > **DNS** > **Response Policy Zones**.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor**:
    a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Infoblox DNS FW Credentials)
Username	User created for SOAR on Infoblox DNS FW
Password	API Password for the key created for SOAR on Infoblox DNS FW
Private Key	Empty

    b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Infoblox DNS Firewall integration on SOAR
Type	Infoblox DNS Firewall
Address	Address of the integration (in the following format: https://192.168.2.53)

Configuration	Specify the following configuration parameters:
	
	``
	# Name of View under which rp_zone is located.
	view=default
	# Name of Response Policy Zone that SOAR will
	write block rules
	rp_zone=mitigated.local
	# Default name and value of extensible attrbute
	which SOAR uses to write comment for block
	extensible.attribute.name=
	extensible.attribute.value=
	# IP address that SOAR uses to substitute in
	DNS A records.
	substitute.ip.address=127.0.0.1
	#proxy.id=5442
	``

Credential	Name of the credential set created on step 2 (For example, Infoblox DNS FW Credentials)
Trust Invalid SSL Certificates	Select this if Web UI's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration.

5. Click **Test** to test the integration.

6. Click **Save** to complete the integration.

Additional Notes

Infoblox DNS Firewall allows blocking IP and host with only one rule type (either No Data or No Such Domain). If you try to block an IP or host that already got blocked with another rule type, you might get an error.

Integration Guide for Invictus USTA ThreatIntelligence

Integration Overview

Invictus USTA is a threat intelligence service which delivers cyber-threat insights in real time.

Integration Capabilities

- Ingest Threat Intelligence Feed as Alert
- Check Identity Leak
- Check Stolen Client Account
- Check Domain Info
- Check Hash Info
- Check IP Info
- Check URL Info
- Submit Bad Sender
- Submit Referer URL

Use Case: Blocking malicious URLs and IPs before they harm

ArcSight SOAR integrates with USTA intelligence feed to block malicious entities on your perimeter protection before they harm.

Use Case #2: Investigating Fraud and ID Theft

SOAR integrates with USTA Threat Intelligence to investigate fraud cases, possible ID theft, and cases of client account compromises.

Configuration

Prerequisites

- Access to https://usta01.invictuseurope.com/api/ (443/tcp port) for SOAR to connect to USTA API
- An API Key for SOAR to connect to Invictus USTA API

Configuring Invictus USPA

Invictus USTA requires no specific configuration.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor**:

   a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example,Invictus USTA Credentials)
Username	Empty
Password	Empty
Private Key	API Key obtained from Invictus USTA platform

   b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

Configuring Invictus USTA as Alert Source

1. Click **Configuration** > **Alert Source** > **Create Alert Source Configuration**.
2. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Invictus USTA Alert Source on SOAR
Type	USTA
Address	Address of the Invictus USTA Threat Intelligence Service (https://usta01.invictuseurope.com/api/)
Alert Severities	Mapping of alert severity values to SOAR incident severities

Parameter	Value

Configuration	Specify the following configuration parameters:
	
	``
	# Ignore events older than specified date. If empty,
	date based filtering is disabled.
	# Example: filterOlderThanDate=2017-01-01
	filterOlderThanDate=2020-01-10
	# Integration ID of the proxy integration to use
	when connecting to current source.
	# If not provided, SOAR will try to use a direct
	connection.
	#proxy.id=5523
	``

Credential	Name of the credential set just created. (For example, Invictus USTA Credentials)
Trust Invalid SSL Certificates	Select this if Web UI's certificate is self-signed or is not recognized by browsers
Visible Alert Fields	Define the alarm fields to be displayed on Incident Management Service Desk

3. Click **Test** to test the integration.

4. Click **Save** to complete the integration.

Configuring Invictus USTA as Integration

1. Click **Configuration** > **Integrations** > **Create Integration**.

2. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Invictus USTA integration on SOAR
Type	USTA
Address	Address of the Invictus USTA Threat Intelligence Service (https://usta01.invictuseurope.com)
Configuration	Specify the following configuration parameters: ``` # Integration ID of the proxy integration to use when connecting to current source. # If not provided, SOAR will try to use a direct connection. #proxy.id=5523#proxy.id=5523 ```
Credential	Name of the credential set created on step 2 (For example, Invictus USTA Credentials)
Trust Invalid SSL Certificates	Select this if Web UI's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration.

3. Click **Test** to test the integration.

4. Click **Save** to complete the integration.

Additional Notes

USTA permits connection requests from specific network addresses for each customer. Hence, make sure to check the access permission by USTA before integration.

Integration Guide for IPInfo

Integration Overview

IPinfo is a solution for IP data which offers both free and paid API tokens to put IP geolocation, ASN, IP to company, mobile carrier, and many more.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with IPinfo:

- IP Query

Configuration

Prerequisites

- You must have access to HTTPS as ArcSight SOAR connects to IPinfo through this service.
- IPinfo requires an API key for access.

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor** form:

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, IPinfo Credential).			Access token

   a. Click **Configuration** > **Integrations** > **Create Integration**.
   b. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of the integration.
Type	IPinfo.io
Address	Address of the integration (the format should be https://ipinfo.io).

Parameter	Value
Configuration	Specify the following configuration parameters:
	<table><tr><td>proxy.id</td><td>ID of the Proxy integration if you access ipinfo.io through a web proxy device. For example: proxy.id = 12345 .</td></tr></table>
Credential	Credential that has been defined for this integration under the **Credentials** menu.
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

c. Click **Save** to save the integration definition.

d. Navigate to **Configuration**>**Customization Library** and edit **IPinfo Advanced Action Script Default Template**.

e. Select the integration that you have added to **Integrations** menu.

f. Click **Save** to complete the integration.

g. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

a. **IP Query**
   Enrichment capability for retrieving information regarding an IP.

   The following table presents the **IP Query** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
IP	Network address to be queried from IPInfo .	Network Address	Yes	Yes
Do not Use Cache	SOAR does not use cached results if this box is checked.	Boolean	N/A	No

Output:

Case Scope:

Enrichment	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

Field	Value
anycast	true
city	Mountain View
country	US
hostname	dns.google
ip	8.8.8.8
loc	37.4056,-122.0775
org	AS15169 Google LLC
postal	94043
region	California
timezone	America/Los_Angeles

Integration Guide for Jira

Integration Overview

Jira is an ITSM service that provides issue management to users.

Unlike our other plugins, this plugin consists of two modules. One was developed as a custom script in SOAR to perform actions on Jira, and the other as an add-on in Jira to perform actions on the SOAR product. We aimed that both products keep each other informed of certain changes on each other. SOAR is using Jira API to perform operations on Jira, and Jira is using our newly developed SOAR API to perform operations on SOAR through the add-on we developed. Issue creation must be initiated with SOAR, so we can mark the issue and track it both sides.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Jira:

- Create Issue
- Send Comment
- Update Issue
- Update Issue Status

Prerequisites

You must have access to HTTPS as the ArcSight SOAR connects to Jira API through this service and Jira connects to SOAR through this service.

Configuration

Configuring SOAR

1. Navigate to **ITOM Management** and click the **Three dots** button for deployment and select **Reconfigure**.

2. Click **SOAR** tab. On the **REST API** fields, specify values for one of them and keep a note of it, as shown in the following figure:
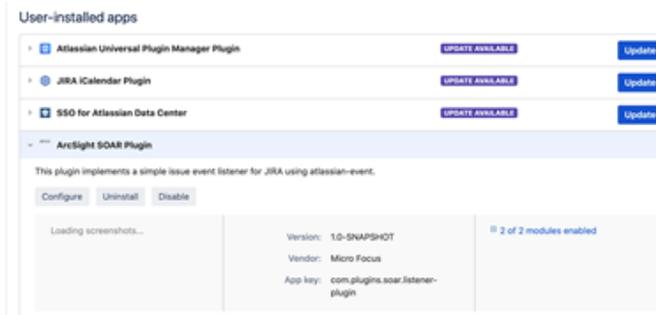
> **Note**: You can note down the **Client Id Suffix** and **Client Secret** values to be used later.

3. Click **Save**

4. Navigate to **SOAR** application and click **Configuration** > **Credentials** > **Create Credential**.

5. Specify the following parameter values in the **Credential Editor:**

Parameter	Value
Type	Internal credential.
Name	Display name of credential set (for example, Jira Credentials)
Username	Jira User Username.
Password	Jira User Password.
Private Key	

6. Click **Save**.

7. Click **Configuration** > **Lists** > **Create List**. Give the list a name (for example, jiraLookup).

> **Note**: SOAR is going to map SOAR cases and Jira issues on this list for both sides.

8. Click **Save**

9. Click **Configuration** > **Integration** > **Create Integration**

10. Specify the following parameter values in the **Configuration Form:**

Parameter	Value
Name	Display name of Jira integration on SOAR.
Type	Jira
Address	Address of the integration (for example, https://192.168.200.231:8080).
proxy.id	ID of the Proxy integration if accessing the jira service through a web proxy device. For Example: proxy.id = 12345.
list.name	Parameter must be equal to list name that is given at step 8. (for example, list.name=jiraLookup).

Parameter	Value
Credential	Name of the credential set created on step 5(for example, Jira Credentials).
Trust Invalid SSL Certificates	Select this if service's certificate is self-signed or is not recognized by browsers.
Required Approval From	Select users from the list who can provide approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

11. Click **Save**.

12. Navigate to **Configuration** > **Customization Library** > **Open Jira Script**.

13. Select integration that is newly created in the **Integrations** field.

14. Click **Save** to complete the integration.

15. Click **Test**, an **Integration Successful** message is displayed if the address and credential are valid.

Configuring Jira

1. Navigate to **Jira Administration**<**Manage apps**.

2. Click **Upload app** and choose the Jira add-on jar file that is provided. After the installation completion, the plugin is visible in the **User-installed** apps.

   > **Note**:You can also download the Jira add-on jar file from Marketplace.

3. Click **Configure**. Specify the values for **Base URL**, **Client ID**, **Client Secret** (as noted during creating an API user in **Configuring SOAR** part) and SOAR username (SOAR needs a JIRA user to access Jira service).

4. Click **Save**.

   > **Note**: Now you can start creating issue on Jira by **Create Issue** capability on SOAR.

Capabilities

1. **Create Issue**
   Action capability for creating issue on Jira.

   The following table presents the **Create Issue** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Project Key	Key of the project that you want to create issue in it.	Text	No	Yes
Issue Type	Type of the issue.	Text	No	Yes
Summary	Summary of the issue.	Text	No	Yes
Description	Description of the issue.	Text	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

2. **Send Comment**
   Action capability for sending comment to related issue.

   The following table presents the **Send Comment** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Comment	Comment that you want to add to the issue.	Text	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

3. **Update Issue**
   Action capability for updating attributes of the issue

   The following table presents the **Update Issue** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Summary	Summary of the issue.	Text	No	No
Description	Description of the issue.	Text	No	No
Assignee	Assignee of the issue.	Text	No	No
Priority	Priority of the issue.	Text	No	No

Output:

Case Scope: N/A

Human Readable Output: N/A

4. **Update Issue Status**
   Action capability for updating status of the issue.

   The following table presents the **Update Issue Status** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Status	Status of the issue	ComboBox (Elements of the combobox are changeable by the script code)	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

> **Note**: We are supporting **Update Status**, **Update Severity**, **Update Description**, **Update Subject** and **Add Comment** capabilities through SOAR application. If the Jira user changes any of the related items in the Jira issue, and if that issue description contains SoarCaseId then the prepared API requests are sent to SOAR.
>
> SOAR then adds the SOAR CaseId into description-field during the creation of the Jira Issue. The Add-On uses this SoarCaseId for SOAR API requests.

Integration Guide for JDBC(Database) Server

Integration Capabilities

ArcSight SOAR has the following integration capability with database servers:

- JDBC Query

Use Case: Querying HR Database

With this integration, while investigating an incident SOAR can run a query on HR database to see if they are logged on the user on a suspicious endpoint. This can either be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- A database listener or service for SOAR to access.
- Create a DB user account for SOAR to run the SQL queries.

Configuring Database Server

Please contact database administrator for user account and access permissions.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor**:

a. Internal Credential

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, JDBC Credentials)
Username	User account that was configured on database server
Password	Password for user account that was configured on database server
Private Key	Empty

b. Credential Store

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store.

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Database Server integration on SOAR
Type	Database Server
Address	Address of the integration (in the format jdbc:driverName://192.168.3.10:5432/databaseName).

Configuration	Specify the following configuration parameters:
	
	``
	# For MySQL: db.driverClass=com.mysql.jdbc.Driver
	# For Oracle: db.driverClass=oracle.jdbc.OracleDriver
	# For PostgreSQL: db.driverClass=org.postgresql.Driver
	# For MSSQL Server:
	db.driverClass=com.microsoft.sqlserver.jdbc.SQLServerDriver
	db.driverClass= db.driverClass=org.postgresql.Driver
	# Absolute path where you put the JDBC driver's JAR file.
	db.driverPath=
	# configure how far (in minutes) into the past this enrichment will look.
	cache.reusing.duration=30
	``

Credential	Name of the credential set created on step 2. (For example, Database Server Credentials).
Trust Invalid SSL Certificates	Select this if device's certificate is self-signed or is not recognized by browsers
Require Approval from	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration.

5. Click **Test**. The following pop up will be displayed if your credential and address are valid.

6. Click **Save** to complete integration.

Integration Guide for Juniper SRX Firewall

Integration Overview

SOAR uses Juniper SRX Firewall to block IP addresses on the network perimeter using the incident scopes.

Integration Capabilities

Action

- Block
- Custom Script

Configuration

Configuring Juniper SRX Firewall

- Access to SSH as SOAR connects to Juniper SRX Firewall integration using SSH
- A SOAR user with admin role

Configuring SOAR

1. Click **Configuration** > **Integrations** > **Create Integration**.
2. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of the integration
Type	Juniper SRX Firewall
Address	Address of the integration (in the following format: 1.1.1.1 or abc.example.com)
Credential	Name of the credential set created on step 2 (For example, FortiMail Credentials)

Parameter	Value
Trust Invalid SSL Certificates	Select this if Integrations's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

> **Note:** You might have to review the integration actions defined and executed through the Juniper SRX Firewall related custom scripts in SOAR.

3. To find the following custom scripts, navigate to **Configuration** > **Custom Scripts**.

- Juniper SRX Firewall Availability Check Default Template

- Juniper SRX Firewall SSH Device Action (Block) Default Template

4. Click **Test** to test the integration.
5. Click **Save** to complete the integration.

Integration Guides for Kannel SMS Gateway

Integration Overview

Kannel is an open source SMS Gateway which is used widely for sending in either single or bulk SMS(Short Message Service). Kannel links HTTP based services to various SMS centers using various protocols.

Integration Capabilities

Supported Action Capabilities

Kannel SMS Gateway allows user notifications using SMS messages which was set when creating the Playbook involving this integration.

Configuration

Configuring Kannel SMS Gateway

- Configure the integration to send SMS messages.

Configuring SOAR

Following are the steps to create the integration:

1. Navigate to **Configuration** > **Parameters**.
2. Configure **SMS Device** to be used as the ID of Kannel SMS Gateway integration.
3. To configure the integration, navigate to **Configuration** > **Integrations**.
4. Specify the following parameter values in the **Integration Editor**:

Parameter	Value
Name	Display name of Kannel SMS Gateway integration on SOAR
Type	Kannel SMS Gateway
Address	Address of the integration (in the following format: 1.1.1.1:1234)

Parameter	Value
Configuration	sms.sender=<Specify the value configured in the SMS Device field>
Credential	Name of the credential set created on step 2
Trust Invalid SSL Certificates	Select this if Integrations's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

5. Click **Test** to test the integration.
6. Click **Save** to complete the integration.

Integration Guide for Kaspersky Security Center

Integration Overview

ArcSight SOAR is capable of communicating with Kaspersky Security Center through WinRM and Powershell to block hashes, add tags to hosts, run tasks, move hosts to groups and retrieve information about various management objects.

Integration Capabilities

- Block (blacklist) SHA-256 or MD5 hash, with rollback support
- Add tag to host, with rollback support
- Move host to group
- Run task
- Retrieve host information

Configuration

Configuration on Kaspersky Security Center

- To define a Kaspersky Security Center installation as an integration on your

SOAR, following integration specific configuration should be performed.

- SOAR should be able to access the server with Kaspersky Security Center through WinRM on the network; usually with TCP port 5985 or 5986 (if SSL is enabled on WinRM). See WinRM Integration Guide for details on how to configure WinRM access.
- A local or domain administrator user account is required execute various capabilities.
- 32-bit version of Windows Scripting Host (which is available on a default Windows installation) is required to execute built-in scripts, which is usually located at C:\Windows\SYSWOW64\cscript.exe.

Configuring SOAR

- While creating this integration via Integrations tab of Configuration menu:
- Name: Display name of the integration.

- Address: Address of the integration. Format of the address should be

IP, IP:port, dns.hostname.localnet, or dns.hostname.localnet:port for HTTP;

or prefixed with https:// if HTTPS/SSL listener was enabled on WinRM.

- Credential: Credential that has been defined for this integration under the Credentials

menu.

Optional configuration

- `blockhash.categoryname:` Category name to add block hashes into; if unspecified defaults to SOAR. If specified category name doesn't exist, it will be automatically created.
- `path.cscriptexe:` Location of the 32-bits version of the cscript.exe on server.

If unspecified, defaults to "C:\\Windows\\SysWOW64\\cscript.exe".

> Note: The backslashes must be escaped and double-backslash is required.

Overriding built-in scripts

SOAR allows overriding built-in scripts using Customization Library. Create a new customization of **Basic plugin script**, take note of its ID, and set the value of the script you'd like to override in the integration configuration by specifying its identifier as specified below:

Parameter Name	Description
enrichment.gettasknames	Retrieve names of tasks available for Run task capability
enrichment.getgroupnames	Retrieve names of groups available for Move host to group capability
enrichment.gettagnames	Retrieve names of tags available for Add tag to host capability
enrichment.hostinfo	Retrieve host information enrichment script
execute.blockhash	Block hash capability
rollback.blockhash	Rollback block hash capability
execute.addtag	Add tag capability
rollback.addtag	Rollback add tag capability
execute.movesystem	Move host to group capability
execute.runtask	Run task capability

Important points

- When these parameters are specified, built-in scripts will be ignored and the customization with specified ID will be used instead as the script. All scripts should target Windows Scripting Host with Javascript language, unless a different/compatible interpreter is specified in path.cscriptexe parameter in integration configuration. See [https://support.kaspersky.com/9291](Kaspersky Enterprise Security Administration Kit Automation10) for reference on using its COM/ActiveX API.

- SOAR's implementation is sensitive to the expected output of these scripts; overriding a capability with a script that doesn't write expected output to stdout may break existing functionality.

- Scripts are automatically evaluated as StringTemplate and various parameters are injected into the template for block hash, run task, move host into group, add tag and host information capabilities. See built-in scripts below for example usage and [http://www.stringtemplate.org](String Template Website) for more details on how to make use of the ST engine.

Example:

4214 is the ID of the customization to override this capability.

execute.runtask=4214

Built-in Tasks

Get Task Names

```
function obj(name) {
return new ActiveXObject("klakaut.KlAk" + name);
}
try {
var oConnectProps = obj("Params"), oAdmServer = obj("Proxy"),
oSrvView = obj("SrvView"),
oTasks = obj("Tasks2"), item, enumObj;
oConnectProps.Add("Address", "127.0.0.1:13291");
oAdmServer.Connect(oConnectProps);
oTasks.AdmServer = oSrvView.AdmServer = oAdmServer;
enumObj = new Enumerator(oTasks.EnumTasks(-1));
WScript.Echo('[OK] [BEGIN]');
for (; !enumObj.atEnd(); enumObj.moveNext()) {
item = enumObj.item();
WScript.Echo(item.item('TASK_UNIQUE_ID') + '=' + item.item('DisplayName'));
}
WScript.Echo('[END]');
```

```
} catch (e) {
WScript.Echo("[Error] " + e.number + " occured !!! " + e.description);
}
```

Get Group Names

```
function obj(name) {
return new ActiveXObject("klakaut.KlAk" + name);
}
function EnumerateGroups(oSubgroupsEnum) {
var enumObj = new Enumerator(oSubgroupsEnum);
for (;!enumObj.atEnd();enumObj.moveNext()) {
var oObj = enumObj.item();
WScript.Echo(oObj.Item("id") + '=' + oObj.Item("name"));
if (oObj.Check("groups")) {
EnumerateGroups(oObj.Item("groups"));
}
}
}
try {
var oConnectProps = obj("Params"), oAdmServer = obj("Proxy"),
oGroups = obj("Groups");
oConnectProps.Add("Address", "127.0.0.1:13291");
oAdmServer.Connect(oConnectProps);
oGroups.AdmServer = oAdmServer;
WScript.Echo('[OK] [BEGIN]');
EnumerateGroups(oGroups.GetSubgroups(oGroups.GroupIdGroups, 0));
WScript.Echo('[END]');
} catch (e) {
WScript.Echo("[Error] " + e.number + " occured !!! " + e.description);

}
```

Get Tag Names

```
function obj(name) {
return new ActiveXObject("klakaut.KlAk" + name);
}
try {
var oConnectProps = obj("Params"), oAdmServer = obj("Proxy"),
oTagsControl = obj("TagsControl"), oProps = obj("Params"), oTags,
enumObj;
oConnectProps.Add("Address", "127.0.0.1:13291");
```

```
oAdmServer.Connect(oConnectProps);
oTagsControl.AdmServer = oAdmServer;
oTagsControl.Prop("ListName") = "HostsTags";
oTags = oTagsControl.GetAllTags(oProps);
WScript.Echo('[OK] [BEGIN]');
if (oTags != null) {
enumObj = new Enumerator(oTags);
for (; !enumObj.atEnd(); enumObj.moveNext()) {
WScript.Echo(enumObj.item() + "=" + enumObj.item());
}
}
WScript.Echo('[END]');
} catch (e) {
WScript.Echo("[Error] " + e.number + " occured !!! " + e.description);
}
```

Host Information Enrichment

```
function obj(name) {
return new ActiveXObject("klakaut.KlAk" + name);
}
function ip2long(IPaddress) {
var ip = IPaddress.match(/^(\d+)\.(\d+)\.(\d+)\.(\d+)$/);
return ip ? (+ip[1] << 24) + (+ip[2] << 16) + (+ip[3] << 8) + (+ip[4]) : null;
}
function long2ip(l) {
return ((l >> 24) & 255) + "." + ((l >> 16) & 255) + "." + ((l >> 8) & 255) +
"." + (l & 255);
}
function coll() {
var ret = obj("Collection"), len = arguments.length, args = arguments;
if (len == 1) {
args = arguments[0].split('|');
len = args.length;
}
ret.SetSize(len);
for (var i=0; i<len; i++) {
ret.SetAt(i, (arguments.length == 1 ? "KLHST_WKS_" : "") + args[i]);
}
return ret;
}
function g(a, e) {
var r = e.item('KLHST_WKS_' + a);
if (r === undefined) {
r = '';
}
return r;
```

```
}
var rtpState = ["Unknown", "Stopped", "Suspended", "Starting", "Running",
"Running (Maximum protection)", "Running (Maximum speed)",
"Running (Recommended settings)", "Running (Custom settings)",
"Failure"];
function getStatus(v) {
var r = [];
if ((v & 1) == 1) {
r.push("Visible");
}
if ((v & 4) == 4) {
r.push("Agent:Installed");
}
if ((v & 8) == 8) {
r.push("Agent:Alive");
}
if ((v & 16) == 16) {
r.push("Real-Time-Protection:Installed");
return r.join(",");
}
try {
var oConnectProps = obj("Params"), oAdmServer = obj("Proxy"),
oHosts = obj("Hosts"), c=0;
oConnectProps.Add("Address", "127.0.0.1:13291");
oAdmServer.Connect(oConnectProps);
oHosts.AdmServer = oAdmServer;
var fieldsToReturn = "LAST_VISIBLE|STATUS|RTP_STATE|LAST_UPDATE|LAST_FULLSCAN|
WINHOSTNAME|WINDOMAIN|OS_NAME|OS_VER_MAJOR|OS_VER_MINOR|IP_LONG|PRODUCT_TAG_
NAME";
var ftr = fieldsToReturn.split('|');
var enumObj = new Enumerator(oHosts.FindHosts("(KLHST_WKS_IP_LONG=" +
ip2long('%host%') + ")", coll(fieldsToReturn), coll()));
WScript.Echo('[OK]');
for (; !enumObj.atEnd(); enumObj.moveNext()) {
var e = enumObj.item();
WScript.Echo('[' + c++ + ']' +
'LAST_VISIBLE=' + Date.parse(g('LAST_VISIBLE', e)) +
'|LAST_UPDATE=' + Date.parse(g('LAST_UPDATE', e)) +
'|LAST_FULLSCAN=' + Date.parse(g('LAST_FULLSCAN', e)) +
'|WINHOSTNAME=' + g('WINHOSTNAME', e) +
'|WINDOMAIN=' + g('WINDOMAIN', e) +
'|OS=' + g('OS_NAME', e) + ' (' + g('OS_VER_MAJOR', e) + '.' +
g('OS_VER_MINOR', e) + ')' +
'|IP=' + long2ip(g('IP_LONG', e)) +
'|RTP_STATE=' + rtpState[g('RTP_STATE', e)] +
'|STATUS=' + getStatus(g('STATUS', e)) +
'|PRODUCT_TAG_NAME=' + g('PRODUCT_TAG_NAME', e)
```

```
);
}
WScript.Echo("[END] Retrieved information for " + c + " hosts.");
} catch (e) {
WScript.Echo("[Error] " + e.number + " occured !!! " + e.description);
}
```

Block Hash Action Capability

```
var hashes = [%hashes: {h | "%h%"}; separator=", "%];
function obj(name) {
return new ActiveXObject("klakaut.KlAk" + name);
}
try {
var oConnectProps = obj("Params"), oAdmServer = obj("Proxy"),
oCategory = obj("FileCategorizer"), oFields2Return = obj("Collection"),
oSrvView = obj("SrvView");
oConnectProps.Add("Address", "127.0.0.1:13291");
oAdmServer.Connect(oConnectProps);
oCategory.AdmServer = oSrvView.AdmServer = oAdmServer;
oFields2Return.SetSize(2);
oFields2Return.SetAt(0, "id");
oFields2Return.SetAt(1, "name");
var enumObj = new Enumerator(oSrvView.GetChunkAccessor
('customcategories',
'(name = "*")', oFields2Return, obj("Collection"))), catFound = null;
for (; !enumObj.atEnd(); enumObj.moveNext()) {
var item = enumObj.item();
if (item.item('name') === '%categoryname%') {
catFound = item.item('id');
}
// dump("", "", item, false);
// dump("", "", oCategory.GetCategory(item.item('id')), false);
}
var oCatToAdd, oInclProps, i, oCatProps = obj("Params"), oCatData = catFound ?
oCategory.getCategory(catFound) : null, oInclusions = catFound ?
oCatData.Item('inclusions') : obj("Collection");
for (i=0; i<hashes.length; i++) {
oInclProps = obj("Params");
oInclProps.Add('ex_type', 3);
oInclProps.Add(hashes[i].length == 32 ? 'str' : 'str2', hashes[i]);
oInclProps.Add('str_op', 0);
oInclusions.SetSize(oInclusions.Count + 1);
oInclusions.setAt(oInclusions.Count - 1, oInclProps);
}
if (!catFound) {
oCatProps.Add('name', '%categoryname%');
```

```
oCatProps.Add('CategoryType', 0);
oCatProps.Add('inclusions', oInclusions);
oCatToAdd = oCategory.CreateCategory(oCatProps);
WScript.Echo("[OK] [CREATED] Added " + hashes.length +
' hashes to newly created category: %categoryname%');
} else {
oCategory.UpdateCategory(catFound, oCatData);
WScript.Echo("[OK] [UPDATED] Added " + hashes.length +
' hashes to existing category: %categoryname% its current size is: '
+ oInclusions.Count);
}
} catch (e) {
WScript.Echo("[Error] " + e.number + " occured !!! " + e.description);
}
```

Rollback of block hash capability

```
var hashes = [%hashes: {h | "%h%"}; separator=", "%];
function obj(name) {
return new ActiveXObject("klakaut.KlAk" + name);
}
try {
var oConnectProps = obj("Params"), oAdmServer = obj("Proxy"),
oCategory = obj("FileCategorizer"), oFields2Return = obj("Collection"),
oSrvView = obj("SrvView");
oConnectProps.Add("Address", "127.0.0.1:13291");
oAdmServer.Connect(oConnectProps);
oCategory.AdmServer = oSrvView.AdmServer = oAdmServer;
oFields2Return.SetSize(2);
oFields2Return.SetAt(0, "id");
oFields2Return.SetAt(1, "name");
var enumObj = new Enumerator(oSrvView.GetChunkAccessor('customcategories',
'(name = "*")', oFields2Return, obj("Collection"))), catFound = null;
for (; !enumObj.atEnd(); enumObj.moveNext()) {
var item = enumObj.item();
if (item.item('name') === '%categoryname%') {
catFound = item.item('id');
}
}
if (!catFound) {
WScript.Echo("[OK] [DOESNTEXIST] Category %categoryname% doesn't exist,
no need to remove anything.");
} else {
var oCatData = oCategory.getCategory(catFound),
oInclusions = oCatData.Item('inclusions'),
oNewInclusions = obj("Collection"), i, j, k=0;
for (j=0; j<oInclusions.Count; j++) {
```

```
for (i=0; i<hashes.length; i++) {
var incl = oInclusions.Item(j);
if (incl.Item('str') !== hashes[i] && incl.Item('str2') !== hashes[i]) {
oNewInclusions.SetSize(oNewInclusions.Count + 1);
oNewInclusions.setAt(oNewInclusions.Count - 1, incl);
} else {
k++;
}
}
}
oCatData.Item('inclusions') = oNewInclusions;
oCategory.UpdateCategory(catFound, oCatData);
WScript.Echo("[OK] [UPDATED] Removed " + k + " of " + hashes.length +
' hashes from category: %categoryname% its current size is: ' +
oNewInclusions.Count);
}
} catch (e) {
WScript.Echo("[Error] " + e.number + " occured !!! " + e.description);
}
```

Add tag to host capability

```
function obj(name) {
return new ActiveXObject("klakaut.KlAk" + name);
}
function ip2long(IPaddress) {
var ip = IPaddress.match(/^(\d+)\.(\d+)\.(\d+)\.(\d+)$/);
return ip ? (+ip[1] << 24) + (+ip[2] << 16) + (+ip[3] << 8) + (+ip[4]) : null;
}
try {
var oConnectProps = obj("Params"), oAdmServer = obj("Proxy"),
oTagsControl = obj("TagsControl"), oHosts = obj("Hosts"),
oFields2Return = obj("Collection"), enumObj, taggedHosts = 0;
oConnectProps.Add("Address", "127.0.0.1:13291");
oAdmServer.Connect(oConnectProps);
oTagsControl.Prop("ListName") = "HostsTags";
oTagsControl.AdmServer = oHosts.AdmServer = oAdmServer;
oFields2Return.SetSize(1);
oFields2Return.SetAt(0, "KLHST_WKS_HOSTNAME");
enumObj = new Enumerator(oHosts.FindHosts("(KLHST_WKS_IP_LONG=" +
ip2long('%host%') +
")", oFields2Return, obj("Collection")));
for (; !enumObj.atEnd(); enumObj.moveNext()) {
var oTagArrayItem = obj("Params");
oTagArrayItem.Add("KLTAGS_VALUE", "%tag%");
oTagArrayItem.Add("KLTAGS_SET", true);
var oTagArray = obj("Collection");
```

```
oTagArray.SetSize(1);
oTagArray.SetAt(0, oTagArrayItem);
var oHostsArrayItem = obj("Params");
oHostsArrayItem.Add("KLTAGS_ITEM_ID", enumObj.item().item('KLHST_
WKS_HOSTNAME'));
oHostsArrayItem.Add("KLTAGS_TAGS", oTagArray);
var oHostsArray = obj("Collection");
oHostsArray.SetSize(1);
oHostsArray.SetAt(0, oHostsArrayItem);
var oSetTagsCallProps = obj("Params");
oSetTagsCallProps.Add("KLTAGS_FULL_REPLACE", false);
oTagsControl.SetTags(oHostsArray, oSetTagsCallProps);
taggedHosts++;
}
WScript.Echo("[OK] Added '%tag%' to " + taggedHosts + " hosts.");
} catch (e) {
WScript.Echo("[Error] " + e.number + " occured !!! " + e.description);
}
```

Rollback of Add Tag to Host Capability

```
function obj(name) {
return new ActiveXObject("klakaut.KlAk" + name);
}
function ip2long(IPaddress) {
var ip = IPaddress.match(/^(\d+)\.(\d+)\.(\d+)\.(\d+)$/);
return ip ? (+ip[1] << 24) + (+ip[2] << 16) + (+ip[3] << 8) + (+ip[4]) : null;
}
try {
var oConnectProps = obj("Params"), oAdmServer = obj("Proxy"),
oTagsControl = obj("TagsControl"), oHosts = obj("Hosts"),
oFields2Return = obj("Collection"), enumObj, tagRemovedHosts = 0,
removedTagCount;
oConnectProps.Add("Address", "127.0.0.1:13291");
oAdmServer.Connect(oConnectProps);
oTagsControl.Prop("ListName") = "HostsTags";
oTagsControl.AdmServer = oHosts.AdmServer = oAdmServer;
oFields2Return.SetSize(1);
oFields2Return.SetAt(0, "KLHST_WKS_HOSTNAME");
enumObj = new Enumerator(oHosts.FindHosts("(KLHST_WKS_IP_LONG=" +
ip2long('%host%') + ")", oFields2Return, obj("Collection")));
for (; !enumObj.atEnd(); enumObj.moveNext()) {
var hostId = enumObj.item().item('KLHST_WKS_HOSTNAME');
var oHostIds = obj("Collection");
oHostIds.setSize(1);
oHostIds.SetAt(0, hostId);
var oExistingTagArray = oTagsControl.GetTags(oHostIds, obj("Params"));
```

```
var oTagArray = obj("Collection");
removedTagCount = 0;
for (var i = 0; i < oExistingTagArray.Count; i++) {
var oTagEntry = oExistingTagArray.Item(i);
var oTagValues = oTagEntry.Item("KLTAGS_TAGS");
for (var j = 0; j < oTagValues.Count; j++) {
var tag = oTagValues.Item(j);
if (tag != '%tag%') {
oTagArray.SetSize(oTagArray.Count + 1);
var oTagArrayItem = obj("Params");
oTagArrayItem.Add("KLTAGS_VALUE", tag);
oTagArrayItem.Add("KLTAGS_SET", true);
oTagArray.SetAt(oTagArray.Count - 1, oTagArrayItem);
} else {
removedTagCount++;
}
}
}
var oHostsArrayItem = obj("Params");
oHostsArrayItem.Add("KLTAGS_ITEM_ID", hostId);
oHostsArrayItem.Add("KLTAGS_TAGS", oTagArray);
var oHostsArray = obj("Collection");
oHostsArray.SetSize(1);
oHostsArray.SetAt(0, oHostsArrayItem);
var oSetTagsCallProps = obj("Params");
oSetTagsCallProps.Add("KLTAGS_FULL_REPLACE", true);
oTagsControl.SetTags(oHostsArray, oSetTagsCallProps);
if (removedTagCount > 0) {
tagRemovedHosts++;
}
}
WScript.Echo("[OK] Removed '%tag%' from " + tagRemovedHosts + "
hosts.");
} catch (e) {
WScript.Echo("[Error] " + e.number + " occured !!! " + e.description);
}
```

Move system to group capability

```
function obj(name) {
return new ActiveXObject("klakaut.KlAk" + name);
}
function ip2long(IPaddress) {
var ip = IPaddress.match(/^(\d+)\.(\d+)\.(\d+)\.(\d+)$/);
return ip ? (+ip[1] << 24) + (+ip[2] << 16) + (+ip[3] << 8) + (+ip[4]) : null;
}
try {
```

```
var oConnectProps = obj("Params"), oAdmServer = obj("Proxy"),
oHosts = obj("Hosts"), oFields2Return = obj("Collection"), enumObj,
hostsToMove = obj("Collection");
oConnectProps.Add("Address", "127.0.0.1:13291");
oAdmServer.Connect(oConnectProps);
oHosts.AdmServer = oAdmServer;
oFields2Return.SetSize(1);
oFields2Return.SetAt(0, "KLHST_WKS_HOSTNAME");
enumObj = new Enumerator(oHosts.FindHosts("(KLHST_WKS_IP_LONG=" +
ip2long('%host%') + ")", oFields2Return, obj("Collection"))));
for (; !enumObj.atEnd(); enumObj.moveNext()) {
hostsToMove.SetSize(hostsToMove.Count + 1);
hostsToMove.SetAt(hostsToMove.Count - 1,
enumObj.item().item('KLHST_WKS_HOSTNAME'));
}
oHosts.MoveHostsToGroup(parseInt('%group%'), hostsToMove);
WScript.Echo("[OK] " + hostsToMove.Count + " hosts moved to group
#%group%");
} catch (e) {
WScript.Echo("[Error] " + e.number + " occured !!! " + e.description);
}
```

Run task capability

```
function obj(name) {
return new ActiveXObject("klakaut.KlAk" + name);
}
try {
var oConnectProps = obj("Params"), oAdmServer = obj("Proxy"),
oTasks = obj("Tasks2"), item, enumObj, taskFound=false;
oConnectProps.Add("Address", "127.0.0.1:13291");
oAdmServer.Connect(oConnectProps);
oTasks.AdmServer = oAdmServer;
enumObj = new Enumerator(oTasks.EnumTasks(-1));
for (; !enumObj.atEnd(); enumObj.moveNext()) {
item = enumObj.item();
if (item.item('TASK_UNIQUE_ID') == '%task%') {
oTask = oTasks.GetTask(parseInt('%task%'));
oTasks.RunTask(parseInt('%task%'));
taskFound = oTask;
}
}
WScript.Echo(taskFound ? '[OK] Task #%task%:' + taskFound.item
('DisplayName') +
' successfully started.' : '[ERROR] Specified task #%task% was not found.');
} catch (e) {
```

```
WScript.Echo("[Error] " + e.number + " occured !!! " + e.description);
}
```

Integration Guide for MAY Siber Scop NET

Integration Overview

MAY Siber Scop NET is a NAC platform that provides visibility to any connected device across the network by integrating switches, routers and firewalls.

Integration Capabilities

ArcSight SOAR has the following integration capability with MAY Siber Scop NET:

Block

Use Case: Isolating Mal-behaving PC

With MAY Siber Scop NET integration, while responding an incident ATAR may block malbehaving computers' network access in order to contain the attack and prevent further spread of the attack. Blocking the host can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Currently SOAR supports MAY Siber Scop NET version 7.1.17.
- SOAR connects to MAY Siber Scop NET API via HTTPS. Typically it runs on 443/tcp port. So access to this service is required.
- An API key is required for SOAR to connect to MAY Siber Scop NET.

Configuring MAY Siber Scop NET

Login to MAY Siber Scop NET and create Web service key under **Settings** > **Global Settings** > **Web Service Key** menu.

Configuring SOAR

1. Navigate to **Configuration** > **Credentials** and click **Create Credential**.
2. Fill the Credential Editor form as follows:

a. **Internal Credential:**

Type	Name	Username	Password	Private Key
Internal credential.	Display name of credential set (i.e., MAY Siber Scop NET Credential).	Empty.	Web Service Key you have created for ATAR on MAY Siber Scop NET.	Empty.

b. **Credential Store:**

Type	Name
External credential.	Name of the credential with pull path of the safe on store.

3. Navigate to **Configuration** > **Integrations** and click **Create Integration**.

4. Fill the configuration form as follows:

Parameter	Value
Name:	Display name of MAY Siber Scop NET integration on SOAR.
Type:	MAY Siber Scop NET.
Address:	Address of the integration (the format should be https://1.1.1.1 or https://abc.example.com).
Configuration:	You need to specify the following configuration parameters: ```# Blocked by message customization
$incident. for incident, $rule. for rule ,$alert. for alert
$incident. for incident, $rule. for rule ,$alert. for alert
$incident. for incident, $rule. for rule ,$alert. for alert
$incident.serial$ for incident serial, $incident.subject$ for incident
subject
$rule.id$ for rule id, $rule.name$ for rule name
for customize reasons followings can be uncomment
#block.reason=Blocked by ATAR - $incident.serial$ $incident.subject$
#rollback.reason=Rollbacked by ATAR - $incident.serial$ $incident.subject$```
Credential:	Name of the credential set you've just created on step 2. (i.e., MAY Siber Scop NET Credential).
Trust Invalid SSL Certificates:	Select this if Engine's certificate is self-signed or not recognized by browsers. Not selected.
Require Approval From:	Select user(s) from list to ask her/his approval before executing actions on this integration.
Notify:	Select user(s) from the list to notify when ATAR performs an action on this integration.

5. When you click the **Test** button a success message is displayed.

6. Click **Save** to complete integration.

Integration Guide for McAfee ePolicy Orchestrator

Integration Overview

McAfee ePolicy Orchestrator (ePO) is a management server for McAfee products which are used to protect endpoints from malware and network threats. It provides a centralized management console to simplify and accelerate the security effectiveness with visibility and control from device to cloud.

Integration Capabilities

- SOAR has the following integration capabilities with McAfee ePolicy Orchestrator:
- Assign Policy
- Apply Tag
- Host Information
- Move Host
- Run Task
- Set TIE Reputation

Use Case: Examining suspicious endpoint

With this integration, during the investigation of an incident SOARmay start an on-demand scan on a suspicious endpoint and may force new policy or move host to other place in system tree regarding scan result. This can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Currently SOAR supports McAfee ePolicy Orchestrator version 5.10.
- SOAR connects to McAfee ePolicy Orchestrator API through HTTPS. Typically it runs on 8443/tcp port. So access to this service is required.
- An user account is required for SOAR to connect McAfee ePolicy Orchestrator.

Configuration on McAfee ePolicy Orchestrator

1. Navigate to **User Management** > **Permission** Sets and create a permission set for SOAR with the following permissions:

Endpoint Security Threat Prevention	View and change task settings
McAfee Agent	View and change policy settings
McAfee TIE Reputations	View and change reputations
Queries and Reports	Use public groups.
Systems	Edit System Tree groups and systems & Apply, exclude, and clear tags
System Tree access	Can search on the following nodes and portions of the System
Tree	My Organization & Can access the following nodes and portions of the System

2. View and change policy settings for the products that you want SOAR to chance policies for (for example: Endpoint Security Threat Prevention, Endpoint Security Firewall, Active Response, etc.)

3. Navigate **User Management** > **Users** and create a user with permission set you in previous step.

Configuring SOAR

1. Navigate to **Configuration** > **Credentials** and click **Create Credential.**

2. Fill the Credential Editor form as follows:

   a. **Internal Credential:**

Type	Name:	Username:	Password:	Private Key:
Internal credential.	Display name of credential set (i.e., McAfee ePO Credentials).	Username you have configured on McAfee ePolicy Orchestrator.	Password for the user you have configured on McAfee ePolicy Orchestrator.	Empty.

   b. **Credential Store:**

Type:	Name:
External credential.	Name of the credential with pull path of the safe on store.

3. Navigate to **Configuration** > **Integrations** and click **Create Integration**.

4. Fill the configuration form as follows:

Parameter	Value
Name	Display name of McAfee ePolicy Orchestrator integration on ATAR.
Type	McAfee ePolicy Orchestrator.
Address	Address of the integration (the format should be https://192.168.2.100:8443).

Configuration	You need to specify the following configuration parameters. For the first integration these values can be left as is:
	
	`system.move.autoSort=false`
	`clienttask.run.retryAttempts =`
	`clienttask.run.retryIntervalInSeconds =`
	`clienttask.run.abortAfterMinutes =`
	`clienttask.run.useAllAgentHandlers =`
	`clienttask.run.stopAfterMinutes=`
	`clienttask.run.randomizationInterval =`
	`policy.assignToSystem.resetInheritance=`

Credential	Name of the credential set you've just created on step 2. (i.e., McAfeeePO Credentials).
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed ornot recognized by browsers.
Require Approval From	Select user(s) from list to ask her/his approval beforeexecuting actions on this integration.
Notify	Select user(s) from the list to notify when SOARperforms an action on thisintegration.

5. When you click the **Test** button a success message is displayed.
6. Click **Save** to complete integration.

Integration Guide for McAfee Network Security Platform (IPS)

Integration Overview

McAfee Network Security Platform is an intrusion prevention system (IPS) to identify

malicious network traffic and stops never-before-seen attacks for which no signatures exist.

Integration Capabilities

SOAR has the following integration capabilities with McAfee Network Security Platform:

- Blacklist MD5 Hash
- Quarantine IP address

Configuration

Prerequisites

- Currently SOAR supports McAfee Network Security Platform version 9.2.7.22.
- SOAR connects to McAfee Network Security Platform's API via HTTPS. By default McAfee Network Security Platform REST-API interface works on 443/tcp port. So access permission to this port is required.
- A user account is required for SOAR to connect McAfee Network Security Platform.

Configuration on McAfee Network Security Platform (IPS)

1. Navigate to **Manager** > **Users and Roles** > **Users** and create a user account with Super User role. In order to access API, Super User role is needed.
2. Navigate to **Devices** and note the device/sensor names.

Configuring SOAR

1. Navigate **Configuration** > **Credentials** and click **Create Credential**.

2. Fill the Credential Editor form as follows:

   a. Internal Credential:

Type	Name	Username	Password	Private Key
Internal credential.	Display name of credential set (i.e., McAfee NSP Credentials).	User you have created for SOAR on McAfee Network Security Platform.	Password of the user you have created for SOAR on McAfee Network Security Platform.	Empty.

   b. Credential Store:

Type	Name
External credential.	Name of the credential with pull path of the safe on store.

3. Navigate to **Configuration** > **Integrations** and click **Create Integration**.

4. Fill the configuration form as follows:

Parameter	Value
Name	Display name of McAfee Network Security Platform integration on SOAR.
Type	McAfee Network Security Platform.
Address	Address of the integration (the format should be https://192.168.2.2).
Credential	Name of the credential set you've just created on step 2. (i.e., McAfee NSP Credentials).
Trust Invalid SSL Certificates	Select this if Platform's certificate is self-signed or not recognized by browsers.

Configuration	You need to specify the following configuration parameters.
	
	`# Name of ISP Devices/Sensors. You may write multiple device names separated by '|' character.`
	`SENSOR_NAME=SENSOR1|SENSOR2`
	`#proxy.id=5442`

Require Approval From	Select user(s) from list to ask her/his approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. When you click on the **Test** button a success message is displayed.

6. Click **Save** to complete integration.

Integration Guide for McAfee Web Gateway

Integration Overview

McAfee Web Gateway is a web filtering solution which utilizes both reputation and categorybased filtering and protection against zero-day malware as well.

Integration Capabilities

SOAR has the following integration capability with McAfee Web Gateway:

- Block URL

Use Case: Blocking access to malicious URL

SOAR can integrate with McAfee Web Gateway to block malicious URLs detected while responding an incident. Blocking can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Currently SOAR supports McAfee Web Gateway version 7.7.2.8.0.
- SOAR connects to McAfee Web Gateway's API through HTTPS. By default McAfee Web Gateway REST-API interface works on 4712/tcp port. So access permission to this port is required.
- A user account for SOAR to connect to McAfee Web Gateway.

Configuration on McAfee Web Gateway

1. Navigate to **Accounts** menu and add a new Role to be used for SOAR user. The new role should have at least "Rest-Interface Accessible" permission.

2. Navigate through Accounts menu and add an Internal Administrator Account with the role you have created in previous step.

3. Create a Wildcard Expression List under **Policy** > **Lists**.

4. Create a new rule and enable it under **Policy** > **Rule Sets** > **URL Filtering** menu to use list created in previous step. Rule criteria should be:

URL.Host matches in list ATARBlock

5. Save changes.

Configuration on SOAR

1. Navigate **Configuration** > **Credentials** and click **Create Credential**.

2. Fill the Credential Editor form as follows:

   a. Internal Credential:

Type	Name	Username	Password	Private Key
Internal credential.	Display name of credential set (i.e., McAfee Web GW Credential).	User you have created for SOAR on McAfee Web Gateway.	Password of the user you have created for SOAR on McAfee Web Gateway.	Empty.

   b. Credential Store:

Type	Name
External credential.	Name of the credential with pull path of the safe on store.

3. Navigate **Configuration** > **Integrations** and click **Create Integration.**

4. Fill the configuration form as follows:

Parameter	Value
Name	Display name of McAfee Web Gateway integration on SOAR.
Type	McAfee Web Gateway.
Address	Address of the integration (the format should be 192.168.1.1:4712).

Configuration	You need to specify the following configuration parameters.
	
	`# Use the McAfee Web Gateway management interface to create the`
	`# list in Policy -> Rule set -> URL filtering section. SOAR will use`
	`# specified list name when adding blocked items.`
	`block.list.name=ATARBlock`

Credential	Name of the credential set you've just created on step 2. (i.e., McAfeeWeb GW Credential).
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed ornot recognized by browsers.
Require Approval From	Select user(s) from list to ask her/his approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. On Integration editor, click **Show Additional Parameters** checkbox and set **ConnectionLimit** to "1" . Because of a limitation of McAfee Web Gateway, this value should never be greater than "1".

6. When you click the **Test** button the following popup should be displayed if yourcredential and address is valid.

7. Click **Save** to complete integration.

Integration Guide for Micro Focus Arcsight ESM

See Integrating SOAR with ESM

Integration Guide for Micro Focus ArcSight Intelligence

Integration Overview

Micro Focus ArcSight Intelligence is using unsupervised machine learning to calculate probabilistic risk assessments based on behavioral analytics from millions of events, ultimately generating a short list of high value targets to allow security teams to detect, investigate, and respond to threats that may be hiding in the enterprise before any case occurs.

Integration Capabilities

SOAR has the following integration capabilities with Micro Focus ArcSight Intelligence:

- Ingest Anomalies as Alert
- Get Entity Details

Use Case #1: Case Prioritization

SOAR is integrated with Micro Focus ArcSight Intelligence, to help the prioritization and investigation of case as well as remediation of cases. When an alert comes a new case is created on SOAR's own Case Management Service Desk. SOAR then automatically checks the risk scores of entities and prioritize the case based on these risk scores.

Use Case #2: Mitigating Account Compromise

SOAR ingests anomaly data from ArcSight Intelligence and create an case ticket on its own Case Management Service Desk. With its broad integration portfolio, orchestration, and automation capabilities, SOAR investigates, ascertains the case, and takes necessary actions to prevent the compromise.

Configuration

Prerequisites

- SOAR connects to Micro Focus ArcSight Intelligence API via HTTPS. By default interface works on 443/tcp port. So access permission to this port is required.
- A user account for SOAR to connect to Micro Focus ArcSight Intelligence API.

Configuring ArcSight Intelligence

No specific configuration is needed on Micro Focus ArcSight Intelligence.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor**:

   Internal Credential

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Micro Focus ArcSight Intelligence Credentials)
Username	User you have created for SOAR on Micro Focus ArcSight Intelligence..
Password	Password of the user you have created for SOAR on Micro Focus ArcSight Intelligence.
Private Key	Empty

3. If **use.basic.authentication** configuration parameter value is **False**, we must get the **Client id** and **Client secret** from Intelligence to ensure that the Intelligence Alert Source and Intelligence Integration works as expected.

   > Note: The default value of **use.basic.authentication** parameter is **False**.

   To get Client ID and Client Secret for Micro Focus Intelligence:

   a. Specify the server name on which the Micro Focus Intelligence works from terminal.

   b. Run the following command to get **Client ID** and **Client Secret** from Micro Focus Intelligence:

   ```
   osp-client-id and osp-client-secret : kubectl get secret osp-secret -n
   arcsight-installer-tyoib -o yaml
   ```

   The output is displayed in the following format:

   ```
   data:
   osp-client-id: NTZjODkyYWE3NDMzZThiOTYzZGVkMjE5ZGIzODU3ZDg=
   osp-client-secret:

   ZjRiZDUzODBiZjQ2NTY5MWQ4NDAzMTFhZTJmMjY1ZGJlZGRjOWU0NDhlZmE3ZDhjN2Q5YzJ
   lY2VjMDkzMmExNw==
   ```

c. Decod the **Client ID** and **Client Secret**.

Run the following command to decode the **Client ID** and **Client Secret**:

```
echo 'NTZjODkyYWE3NDMzZThiOTYzZGVkMjE5ZGIzODU3ZDg=' | base64 --decode
echo
'ZjRiZDUzODBiZjQ2NTY5MWQ4NDAzMTFhZTJmMjY1ZGJlZGRjOWU0NDhlZmE3ZDhjN2Q5Yz
JlY2VjMDkzMmExNw==' | base64 --decode
```

d. Add the the **Client ID** and **Client Secret** to the Alert Source / Integration configuration on Micro Focus Intelligence.

Run the following command to add the **Client ID** and **Client Secret**:

```
# Client id that defined in OSP
client.id=id
# Client secret that defined in OSP
client.secret=secret
```

Configuring Micro Focus ArcSight Intelligence as Alert Source

1. Click **Configuration** > **Integrations** > **Create Alert Source**.

2. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Micro Focus ArcSight Intelligence Alert Source on SOAR.
Type	Micro Focus ArcSight Intelligence.
Address	Address of the Micro Focus ArcSight Intelligence server (the format must be https://172.16.11.9).

Para mete r	Value
Confi gurati on	Specify the following configuration parameters:

```
# Tenant id
tenant.id=
# Minimum risk of anomaly, default value is 25. Value must be between 0-100 (lowest
to highest)
min.risk=75
# Scope fields to be extracted from correlated events (field1:CATEGORY:ROLE,
field2:CATEGORY:# CATEGORY is any of: EMAIL_ADDRESS, HASH, HOST, MAC_ADDRESS,
NETWORK_ADDRESS,
# COMPUTER_NAME, UNKNOWN, URL, USERNAME, PROCESS
# ROLE is any of: OFFENDER, IMPACT, RELATED
#
# Note: Field names must start with / character
#
# Example: correlated.scope=/sourcev6:NETWORK_ADDRESS:OFFENDER
# correlated.scope=
# How far (in days) into the past SOAR will look for alerts at the initial sync
task
# If not provided, SOAR will use 14 days by default
days.to.look.back.at.initial.sync=14
# ID of the proxy integration to use when connecting to current source.
# If not provided, SOAR will try to use a direct connection.
#proxy.id=123
# To use basic authentication, use.basic.authentication=true
# If not provided, SOAR will consider this property as false
# If use.basic.authentication is false, client.id and client.secret must be
specified from from Intelligence side for the Intelligence Alert Source /
Integration to work as expected.
use.basic.authentication=false
# Base path of the Interset. We are adding it to end of the URL to access Interset.
interset.context.path=/interset
# Client id that defined in OSP
client.id=id
# Client secret that defined in OSP
client.secret=secret
```

> Note: The Micro Focus Intelligence uses 0 for tenant id by default. The Micro Focus Intelligence - SOAR integration supports different tenants.

Cred ential	Name of the credential set you have created (For example, Micro Focus ArcSight Credentials Credentials)

Para mete r	Value
Trust Invali d SSL Certif icates	Select this if Web UI's certificate certificate is self-signed or is not recognized by browsers
Visibl e Alert Fields	You may define which alarm fields will be displayed on Case Management Service Desk.

3. Click **Save** to complete the integration.
4. Click **Test** to test the integration.

Configuring Micro Focus ArcSight Intelligence as Integration

1. Click **Configuration** > **Integrations** > **Create Integration**.
2. Specify the following parameter values in the **Configuration** form:

Paramete r	Value
Name	Display name of Micro Focus ArcSight Intelligence integration on SOAR.
Type	Micro Focus ArcSight Intelligence.
Address	Address of the Micro Focus ArcSight Intelligence server (the format must be https://172.16.11.9).

Parameter	Value

Configuration	Specify the following configuration parameters:
	
	``# Tenant id
	tenant.id=
	# ID of the proxy integration to use when connecting to current source.
	# If not provided, SOAR will try to use a direct connection.
	#proxy.id=123
	# configure how far (in minutes) into the past this enrichment will look.
	#cache.reusing.duration=20
	# To use basic authentication, use.basic.authentication=true
	# If not provided, SOAR will consider this propert as false
	# If use.basic.authentication is false, client.id and client.secret must be filled
	use.basic.authentication=false
	# Base path of the Micro Focus Intelligence. We are adding it to end of the URL to access Micro Focus Intelligence.
	interset.context.path=/interset
	interset.context.path=/interset
	interset.context.path=/interset
	client.id=id
	# Client secret that defined in OSP
	client.secret=secret```

Credential	Name of the credential set you have created (For example, Micro Focus ArcSight Credentials Credentials)
Trust Invalid SSL Certificates	Select this if Web UI's certificate certificate is self-signed or is not recognized by browsers
Require Approval From	Select user(s) from list to ask her/his approval before executing actions on this integration. Since SOAR only executes enrichments on Interset, leave it empty.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration. Since SOAR only executes enrichments on ArcSight Intelligence, leave it empty.

3. Click **Save** to complete the integration.

4. Click **Test** to test the integration.

Additional Notes

- Get Entity Details enrichment results return latest 1000 records in maximum.

- The following configuration parameters can be used for fine tuning the integration. You must consult SOAR field engineering team before editing them:

  ○ MicroFocusIntelligenceListenerMaxRetrySeconds Micro Focus Interset listener queue max message retry in seconds 1800

○ MicroFocusIntelligenceListenerQueueConcurrency Upper limit of Micro Focus Interset Listener consumer thread count 3

○ MicroFocusIntelligenceSyncPeriod Period in seconds to sync Micro Focus Interset anomalies 60

Capabilities

1. **Get Details**

   Enrichment capability for getting risk score of a given entity and related alert details.

   The following table presents the **Get Details** capability details:

Input Parameter	Description	Type	Scope Rescticted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Entity	Entity to be queried on ArcSight Intelligence.	Network Address Host File Name URL	Yes	Yes
Do not use cache	SOAR does not use cached results if this box is checked.	Checkbox	N/A	No

   Output:

   Case Scope:

Action	Type	Category/ Value
Add Scope Item Property	Integer	Micro Focus Intelligence Entity Risk
Add Scope Item Property	TEXT	Micro Focus Intelligence Entity Hash
Add Scope Item Property	TEXT	Micro Focus Intelligence Entity Type

   Human Readable Output:

a. Risk tab:

b. Alerts tab:

Integration Guide for Micro Focus ArcSight Logger

Integration Overview

ArcSight Logger is a log management solution for compliance, efficient log search, and secure storage.

Integration Capabilities

ArcSight SOAR has the following integration capability with Micro Focus ArcSight Logger:

- Search Query

Use Case: Investigating Cyber-attacks

Integrated with Micro Focus ArcSight Logger, ATAR queires logs collected from various enterprise systems to enrich incident ticket, and improve analyst's understanding of incident.

Configuration

Prereqisites

- Currently SOAR supports Micro Focus ArcSight Logger version 6.3.1.7874.0 and later.SOAR connects to Micro Focus ArcSight Logger API using HTTPS. By default REST-API interface works on 443/tcp port. So access permission to this port is required.

- A user account is required for ATAR to connect Micro Focus ArcSight Logger.

Configuration on Micro Focus ArcSight Logger

- Click **System Admin** > **Users/Groups** > **User Management** and add a user account with **Default Logger Search Group**.

Configuring SOAR

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor**:
   a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, ArcSight Logger Credentials)
Username	User you have created for ATAR on Micro Focus ArcSight Logger.
Password	Password of the user you have created for ATAR on Micro Focus ArcSight Logger.
Private Key	Empty

   b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Micro Focus ArcSight Logger integration on SOAR
Type	Micro Focus ArcSight Logger
Address	Address of the integration (the format must be https://192.168.12.6)

Parameter	Value
Configuration	Specify the following configuration parameters: ``` events.pageLength=10000 # configure how far (in minutes) into the past this enrichment will look. #cache.reusing.duration=20 # local search enabling parameter for Search Query capability. # If this is set false, ATAR will perform searches on all nodes. #local.search.enabled=false # use master session while fetching events from peers for Search Query. # If this is set true, ATAR will use the same session ID while performing # searches on the other nodes. #reuse.master.session=false # peers credential list (if master session won't be shared) # peer address and credential ID values must be separated with : # additional peer-credential pairs must be separated with | #peer.credential.list=1.1.1.1:CredentialId|2.2.2.2:CredentialId ```
Credential	Name of the credential set created on step 2 (For example, ArcSight Logger Credentials)
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or is not recognized by browsers

5. Click **Test** to test the integration.

6. Click **Save** to complete the integration.

Additional Notes

- In order to execute queries on Micro Focus ArcSight Logger, you should create query scripts with **ArcSight Logger Query** type under **Configuration** -> **Customization Library**.

- SOAR extracts scope items on columns defined as Artifact in the query script. For example, `// Artifact: deviceCustomNumber1Label | KEYWORD | RELATED`

Integration Guide for Microsoft Active Directory

Integration Overview

Microsoft Active Directory is an umbrella title for directory-based identity related services that Microsoft developed for the Windows domain networks.

ArcSight SOAR has the following integration capabilities with Microsoft Active Directory:

- Add user to a group
- Remove user from a group
- Lock user acoount
- Get user information
- Get user's groups
- Get group list
- Get group information
- Get computer information
- List computers on domain
- Fetch a domain object

Use Case: Compromised user account

During the investigation of the attack SOAR can ask Microsoft Active Directory the details of theuser account suspicious to be compromised, check the groups account belongs to, locks the account, fetches her/his manager's information and send a notification e-mail to manager if needed.

This can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- SOAR connects to Microsoft Active Directory using LDAPS protocols. Access to 636/tcp
- port is required.
- A domain user account is required for SOAR to connect Microsoft Active Directory.

Configuration on Microsoft Active Directory

- Create a user account on Domain Controller with no password expiry.
- Add this user into "Account Operators" group. Members of this group can manage groups and accounts on domain except domain admins.

Configuring SOAR

1. Navigate **Configuration** > **Credentials** and click **Create Credential**.
2. Fill the Credential Editor form as follows:

   a. Internal Credential:

Type	Name:	Username	Password	Private Key
Internal credential.	Display name of credential set (i.e., Microsoft AD Credentials).	User you have created for SOAR on Microsoft Active Directory (the format should be username@domain).	Password of the user you have created for SOAR on Microsoft ActiveDirectory.	Empty.

   b. Credential Store:

Type	Name
External credential.	Name of the credential with pull path of the safe on store.

3. Navigate to **Configuration** > **Integrations** and click **Create Integration.**
4. Fill the configuration form as follows:

Parameter	Value
Name	Display name of Microsoft Active Directory integration on SOAR.
Type	Microsoft Active Directory.
Address	Address of the integration (the format should be 192.168.2.2:636).

Parameter	Value

Configuration	You need to specify the following configuration parameters.
	
	``
	# SOAR will search objects under LDAP searchbase specified.
	# Format should be "DC=EXAMPLE,DC=COM"
	ldap.searchbase=DC=EXAMPLE,DC=COM
	# LDAP domain should be like "example.com"
	ldap.domain=example.com
	# LDAP NT domain name should be like "EXAMPLE"
	ldap.ntdomain=EXAMPLE
	# Username for LDAP service availability check.
	# SOAR will try to bind LDAP service as this user.
	ldap.checkavailabilityuser=testuser01@example.com
	# configure how far (in minutes) into the past this enrichment will look.
	
	cache.reusing.duration=30
	``

Credential	Name of the credential set you've just created on step 2. (i.e., Microsoft AD Credentials).
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or not recognized by browsers.
Require Approval From	Select user(s) from list to ask her/his approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click on the **Test** button.

6. Click **Save** to complete integration.

Integration Guide for Microsoft Azure Active Directory

Integration Overview

Azure Active Directory (Azure AD) is Microsoft's cloud-based identity and access management service, that helps users to sign-in and access both external and internal resources, for example Microsoft 365, Azure portal, SaaS applications and many more.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Azure Active Directory:

- Add User to Group
- Disable User
- Enable User
- Get User Details
- Get User's Manager
- List Groups
- List User's Groups
- List Users
- Remove User from Group
- Revoke Sign-in Sessions

Prerequisites

You must have access to HTTPS as the ArcSight SOAR connects to Azure Active Directory API through this service.

Configuration

Configuring Microsoft Azure

1. Login to Azure Portal and navigate to **Azure Active Directory** service.

2. Register through the the **App Registrations** tab and create a new registration with the following values:

Name	Supported Account Types	Redirect URL
ArcSight SOAR	Accounts in the organizational directory (Default Directory only - Single tenant)	(Web) https://localhost/soar

3. Click **Add a certificate or secret** and create a new **Client secret**. Add ArcSight SOAR as description and specify the expiry period as 24 months.

4. Note the **Secret Key** value and **Client ID**.

5. Navigate to **API Permissions** and add the following permissions:

Permission Type	Permission	Description
Delegated	Directory Access as user All	Access directory as the signed in user
Application	Directory Read write All	Read and write directory data
Application	User Read write All	Read and write all users' full profiles5.

6. . Click **Yes** to grant admin consent for **Default Directory**

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**.

2. Specify the following parameter values in the **Credential Editor** form.

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, Azure AD Credentials).		Client ID of the application (for example, ArcSight SOAR) that you registered on Azure portal.	Secret Key

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration** form

Parameter	Value
Name	Display name of the integration.
Type	Microsoft Azure Active Directory
Address	Address of the integration (for example, https://graph.microsoft.com/v1.0).
Configuration	Specify the following configuration parameters:

list.name	Tenant ID on Microsoft Azure tenant.id = ff1f0000-c600-4500-0038-9d4000000000
proxy.id	ID of the Proxy integration if you access Microsoft Azure Active Directory through a web proxy device. For example, proxy.id = 12345 .

Parameter	Value
Credential	Credential that has been defined for this integration under the **Credentials** menu.
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Save** to save the integration definition.

6. Navigate to **Configuration**>**Customization Library** and edit **Microsoft Azure Active Directory Advanced Action Script Default Template**.

7. Select the integration that you have added to **Integrations** menu.

8. Click **Save** to complete the integration

9. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Add User to Group**
   Action capability for adding a user to given AD group.

   • Rollback: Yes

   • Duplicate Control: No

   The following table provides the **Add User to Group** action capability details

Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
User	Username to be added to group	Username Email Address Keyword Unknown	Yes	Yes
Group ID	Target group ID	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

2. **Disable User**
   Action capability for disabling user account by blocking the sign-in procedure.

   • Rollback: Yes

   • Duplicate Control: No

   The following table provides the **Disable User** action capability details:

Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
User	Username to be disabled.	Username Email Address Keyword Unknown	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

3. **Enable User**
   Action capability for enabling user account by removing sign-in block.

   • Rollback: Yes

   • Duplicate Control: No

The following table provides the **Enable User** action capability details:

Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
User	Username to be enabled	Username Email Address Keyword Unknown	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

4. **Get User Details**

   Enrichment capability for retrieving user details.

   The following table provides the **Get User Details** enrichment capability details:

Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	User to be queried from Active Directory	Username Email Address Keyword Unknown	Yes	Yes

Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

5. **Get User's Manager**

   Enrichment capability for retrieving user's manager.

   The following table provides the **Get User's Manager** enrichment capability details:

Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	User to be queried for manager's information.	Username Email Address Keyword Unknown	Yes	Yes

   Output:

   Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

   Human Readable Output:

6. **List Groups**

   Enrichment capability for retrieving AD group list.

   The following table provides the **List Groups** enrichment capability details:

Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes

   Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

Name	Description	Group Mail	Created Time	Group Id
QATeam	QA & Test Team		2021-07-23T08:46:54Z	74646e61-6459-4d7d-b9b5-25923adcb841
DevTeam	Development Team		2021-07-23T08:46:32Z	9c381fcf-6d25-47b3-be65-31b6257b26a2
Red Team	Red Team		2021-08-06T13:27:19Z	c94b4b16-fd61-441e-9b01-eaa695c80b64
Blue Team	Blue Team		2021-08-06T13:26:49Z	d2b52062-b52c-4a77-aac4-9ba91423c255

7. List User's Groups

   Enrichment capability for retrieving the list of groups for a specified username.

   The following table provides the **List User's Group** enrichment capability details:

Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	User to be queried for group memberships.	Username Email Address Keyword Unknown	Yes	Yes

   Output:

   Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

   Human Readable Output:

Name	Description	Group Mail	Created Time	Group Id
DevTeam	Development Team		2021-07-23T08:46:32Z	9c381fcf-6d25-47b3-be65-31b6257b26a2
Purple Team	Purple Team		2021-08-06T13:29:48Z	d553b299-17a2-407c-98f9-6dcbbf59ab92

8. List Users

   Enrichment capability for retrieving list of users.

   The following table provides the **List Users** enrichment capability details:

Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes

Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

9. Remove User from Group

Action capability for removing a user from given AD group.

• Rollback: Yes

• Duplicate Control: No

The following table provides the **Remove Users Group action** capability details:

Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
User	Username to be removed from group	Username Email Address Keyword Unknown	Yes	Yes
Group ID	Target group ID	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

10. **Revoke Sign-in Sessions**

Action capability for revoking all the refresh action of the user and session tokens issued to applications, by resetting the **signInSessionsValidFromDateTime** user property to the current date.

This forces the user to sign in to those applications again.

• Rollback: No

• Duplicate Control: No

The following table presents the **Revoke Sign-in Session** enrichment capabilities details:

Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
User	Username to be forced to revoke (terminate) sign-in sessions.	Username Email Address Keyword Unknown	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

Integration Guide for Micro Focus IT Service Manager

Integration Overview

Micro Focus Service Manager is an IT Service Management (ITSM) Tool that uses the Information Technology Infrastructure Library (ITIL) framework to provide a web interface for corporate changes, releases and interactions (request fulfillment) that is supported by a service catalog and Configuration Management Database (CMDB).

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Micro Focus IT Service Manager:

- Close Incident
- Create Incident
- Update Incident

Prerequisites

You must have access to HTTPS as the ArcSight SOAR connects to Micro Focus IT Service Manager API through this service.

Configuration

Configuring Micro Focus IT Service Manager

1. Create a user on IT Service Manager with admin role. This user must be able to and consume the rest APIs of the IT Service Manager.

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential.**
2. Specify the following parameter values in the **Credential Editor** form.

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, Micro Focus IT Service Manager Credentials).	Username of the created user on Micro Focus IT Service Manager.	Password of the created user on Micro Focus IT Service Manager.	

3. Click **Configuration** > **Lists** > **Create List**. The list must have two columns with the type keyword. Add a name to the list and save it. The name of the list is used during integration configuration.
4. Click **Configuration** > **Integrations** > **Create Integration**.
5. Specify the following parameter values in the **Configuration** form.

Parameter	Value
Name	Display name of the integration.
Type	Micro Focus IT Service Manager
Address	URL of the Micro Focus IT Service Manager integration (for example, http://15.113.165.82:13080).
Configuration	Specify the following configuration parameters:
		list.name	List name that is used for mapping ArcSight SOAR cases to Micro Focus IT Service Manager incidents. For example, list.name=mfitsmMapList
		proxy.id	ID of the Proxy integration if you access Micro Focus IT Service Manager through a web proxy device. For example, proxy.id = 12345 .
Credential	Credential that has been defined for this integration under the **Credentials** menu.
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

6. Click **Save** to save the integration definition.
7. Navigate to **Configuration**>**Customization Library** and edit **Micro Focus IT Service Manager Advanced Action Script Default Template**.
8. Select the integration that you have added to **Integrations** menu.

9. Click **Save** to complete the integration.

10. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Create Incident**

   Action capability for creating incident on Micro Focus IT Service Manager

   Rollback : No

   Duplicate Check: Yes

   The following table presents the **Create Incident** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Category	Category information of created incident.	Enum	No	Yes
Description	MF ITSM Incident Description.	Text	No	Yes
Title	Incident Title	Text	No	Yes
Service	Service Type	Enum	No	Yes
Impact	Incident Impact	Enum	No	Yes
Urgency	Incident Urgency	Enum	No	Yes
Status	Incident Status	Enum	No	No
Alert Status	Incident Alert Status	Text	No	No
Area	Incident Area	Text	No	No
Subarea	Incident Subarea	Text	No	No
Assignment Group	Incident Assignee	Text	No	No
Affected CI	Incident Affected CI	Text	No	No
Company	Incident Company	Text	No	No
Phase	Incident Phase	Text	No	No

2. **Close Incident**

   Action capability for closing incident on Micro Focus IT Service Manager.

   Rollback : No

   Duplicate Check: Yes

   The following table presents the **Close Incident** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Solution	Solution Note	Text	No	Yes

3. **Update Incident**

   Action capability for updating incident on Micro Focus IT Service Manager.

   Rollback : No

   Duplicate Check: No

   The following table presents the update incident action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Description	MF ITSM Incident Description.	Text	No	Yes
Title	Incident Title	Text	No	Yes
Service	Service Type	Enum	No	Yes
Impact	Incident Impact	Enum	No	Yes
Urgency	Incident Urgency	Enum	No	Yes
Status	Incident Status	Enum	No	No
Alert Status	Incident Alert Status	Text	No	No
Area	Incident Area	Text	No	No
Subarea	Incident Subarea	Text	No	No
Assignment Group	Incident Assignee	Text	No	No
Affected CI	Incident Affected CI	Text	No	No
Company	Incident Company	Text	No	No
Phase	Incident Phase	Text	No	No

Integration Guide for Micro Focus UCMDB

Integration Overview

Micro Focus Universal Configuration Management Database (UCMDB) generates and maintains a Configuration Management Database of information technology items. It includes a mechanism for automated discovery of IT infrastructure components, such as computers and network devices.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Micro Focus UCMDB:

- Expose CI Information
- Get CI
- Get Related CIs

Prerequisites

You must have access to HTTPS as the ArcSight SOAR connects to Micro Focus UCMDB API through this service.

Configuration

Configuring Micro Focus UCMDB

Create a user with privileges to use REST API. The username and password of the user is used as credential in the ArcSight SOAR.

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**

2. Specify the following parameter values in the **Credential Editor** form:

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, Micro Focus UCMDB Credentials).	Username	Password	

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration** form:

Parameter	Value		
Name	Display name of the integration.		
Type	Micro Focus UCMDB		
Address	URL of UCMDB (ie. https://cms.smax.swdemos.net:8443)		
Configuration	Specify the following configuration parameters:		
	cache.reusing.duration	Configure how far (in minutes) into the past this enrichment will look. For example, cache.reusing.duration=20 .	
	max.result.count	Maximum result count for Get Observed Attack Techniques capability.For example: max.result.count=200	
	proxy.id	ID of the Proxy integration if you access Micro Focus UCMDB through a web proxy device. For example, proxy.id = 12345	
Credential	Credential that has been defined for this integration under the **Credentials** menu.		
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers.		
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.		
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.		

5. Click **Save** to save the integration definition.

6. Navigate to **Configuration**>**Customization Library** and edit **Micro Focus UCMDB Advanced Action Script Default Template**.

7. Select the integration that you have added to **Integrations** menu.

8. Click **Save** to complete the integration.

9. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Expose CI Information**
   Enrichment capability for information related to the CIs of a certain type.

   The following table presents the **CI Enrichment** capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	No	Yes
Layout	The comma seperated columns that are displayed in the output, for example, display_label, name, description, node_role	Text	No	Yes
Type	The CI Type. For example, node, sqlserver, unix.	Text	No	Yes
Column	The value of this column is checked against the value you provided,for example, application_ip or name	Text	No	No
Value	Value, that is going to used during filtering.	ScopeItem	Yes	No

   Output:

   Case Scope:

Action	Type	Category/ Value
Add	Scope Item	Keyword(Related)

   Human Readable Output:

2. **Get CI**
   Enrichment capability for returning details of a CI.

The following table presents the **Get CI** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	No	Yes
ID	CI id. If provided this value will be used regardless of the IP and Type values.	Keyword	Yes	No

Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

3. **Get Related CIs**

   Enrichment capability for returning the details of the CIs related to the specified CI.

   The following table presents the **Get Related CI** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	No	Yes
ID	CI id. If provided this value will be used no matter type or ip provided or not.	Keyword	Yes	Yes
Type	The string that represents the name of a valid configuration item type from the UCMDB. The name of the CI Type can be found inside the CI Type Manager.	Text	No	Yes

Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:
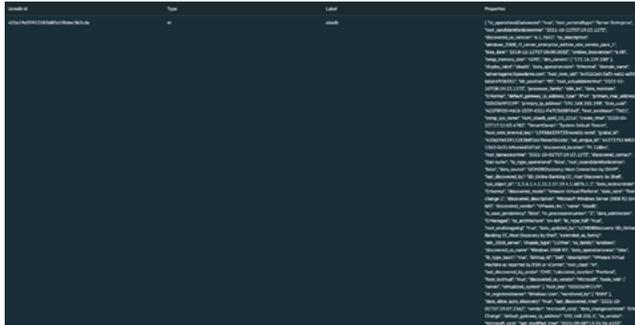
Integration Guide for Microsoft Exchange

Integration Overview

Exchange Server is a mail server developed by Microsoft.

SOAR has the following integration capabilities with Microsoft Exchange Server :

- Delete email
- Mark email
- Quarantine email

Use Case: Deleting already delivered phishing emails

SOAR can follow email inboxes for user's phishing reports and automatically creates an incident record on its service desk. During the investigation of the attack SOAR can extract the sender address and subject and using these values performs a search on Microsoft Exchange Server to mark or delete already delivered malicious messages. This can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- SOAR connects to Microsoft Exchange Web Service API via HTTPS. So access to 443/tcp port is required.
- A user account with impersonation role is required for SOAR to connect Microsoft Exchange.

Configuration on Microsoft Exchange

1. Login to Microsoft Exchange admin center and add a user mailbox for SOAR.
2. Open Exchange Management Shell and give the user Application Impersonation role using the following command:

```
New-ManagementRoleAssignment \
   -Name:<impersonation Assignment Name> \
   -Role:ApplicationImpersonation \
   -User:<account name>
```

Configuration on SOAR

1. Navigate **Configuration** > **Credentials** and click **Create Credential**.
2. Specify the parameter values in the **Credential Editor** form as follows:

    a. Internal Credential:

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (i.e., Microsoft Exchange Credentials).	User you have configured SOAR on Microsoft Exchange (the format should be username@domain).	Password of the user you have configured for SOAR on Microsoft Exchange.	Empty

    b. Credential Store:

    Type: External credential.

    Name: Name of the credential with pull path of the safe on store.

3. Navigate to**Configuration** -> **Integrations** and click **Create Integration**.
4. Specify the parameter values in the **Configuration** form as follows:

Address	Configuration	Credential	Trust Invalid SSL Certificated	Require Approval from	Notify	Require Approval from	Notify

Display name of Microsoft Exchange integration on SOAR.	Microsoft Exchange	Address of the integration (the format should be 192.168.2.8).	You need to specify the following configuration parameters
			
			`requests.impersonation.disable=false`
			
			`requests.cookies.enable=true`
			
			`mail.store.protocol=exchange`
			
			`mail.incoming.pollerperiod=10000`
			
			`mail.incoming.folder=Inbox`
				Name of the credential set you've just created on step 2. (i.e., Microsoft Exchange Credentials).	Select this if certificate used on Exchange Server is self-signed or not recognized by browsers.	Select user(s) from list to ask her/his approval before executing actions on this integration	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Test** to test the integration.
6. Click **Save** to complete integration.

Additional Notes

- To customize warning messages for Quarantine and Mark actions, edit the following parameters under **Configuration** > **Parameters**:
  - MSExchangeMarkWarningText
  - MSExchangeQuarantineWarningText
- To customize the mail folder to be used for Quarantine actions, edit the following parameter under **Configuration** > **Parameters**:
  - MSExchangeQuarantineEMailBox
- In some environments with multiple CAS deployments Exchange uses a request cookie to track the environment. The requests.cookies.enable configuration can help track the cookie so that SOAR won't have any mismatch and Subscription was not found error. It is by default true and should stay that way in most environments.

Integration Guide for Microsoft Office365 Exchange EWS

Integration Overview

Exchange Server EWS provide access to mailbox data stored in Exchange Online, Exchange Online as part of Office 365, and on-premises versions of Exchange starting with Exchange Server 2007, and enable you to manage that information according to the requirements of your organization.

> Note: This is the new version of Microsoft Exchange integration and old one will be phased out.

Users are encouraged to use this integration.

ArcSight SOAR has the following integration capabilities with Microsoft Exchange EWS :

- Block Email Sender
- Delete Email
- Delete Attachment
- Get Attachments
- Get Emails
- Search Emails

Use Case: Deleting already delivered phishing emails

SOAR follows email inboxes for user's phishing reports and automatically creates an incident record on its service desk. During the investigation of the attack ATAR can extract the sender address and subject and using these values performs a search on Microsoft Exchange Server to delete already delivered malicious messages and block malicious senders. This can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- SOAR connects to Microsoft Exchange Web Service API using HTTPS. So access to 443/tcp port is required.

- A user account with the following permissions is required for SOAR to connect MS Exchange EWS Server:
  - ApplicationImpersonation (Authorized to make operations for other users' accounts)
  - MailboxSearch (Authorized to search all mailboxes).

Configuration on Microsoft Exchange

1. Login to Microsoft Exchange Admin Center (For example, https://exchangeserver/ecp) and add a user mailbox for SOAR.

2. Navigate to **Permissions** > **Cloud Migrator Impersonation**, edit and add user account you have created in first step to "Members" to give Account Impersonation permission.

3. Navigate to **Permissions** > **Discovery Management**, edit and add user account you have created in first step to "Members" to give Mailbox Search permission.

Configuring SOAR

1. Navigate **Configuration** > **Credentials** and click **Create Credential**.
2. Fill the Credential Editor form as follows:

   a. Internal Credential:

   Type: Internal credential.

   Name: Display name of credential set (i.e., MS Exchange EWS Credentials).

   Username: User you have configured ATAR on Microsoft Exchange (the format should be username@domain).

   Password: Password of the user you have configured for ATAR on Microsoft Exchange.

   Private Key: Empty.

   b. Credential Store:

   Type: External credential.

   Name: Name of the credential with pull path of the safe on store.

3. Navigate **Configuration** > **Integrations** and click **Create Integration**.
4. Fill the configuration form as follows:

   Name: Display name of Microsoft Exchange EWS integration on ATAR.

   Type: Microsoft Exchange EWS.

   Address: Address of the integration (the format should be outlook.office365.com or 192.168.2.7).

   Configuration : You need to specify the following configuration parameters.

```
# Maximum record number per paginated response. Default value is 1000
page.size=200
# Connect time out in seconds. Default value is 200
connect.timeout=7200
# Request time out in seconds. Default value is 200
request.timeout=7200
# Trash folder name. Default value is Deleted Items
#trash.folder=
# Junk folder name. Default value is Junk Email
#junk.folder=
# Maximum record number per paginated attachment detail response. Default
value is 10
#attachment.page.size=
# Microsoft Exchange Server enrichment API timezone, if not specified GMT
will be used as default
#timezone=
# Maximum number of email id list per request. Default value is 5
#email.id.size=
# Maximum record number per paginated item detail response. Default value
is 10
#email.page.size=
# Maximum email item limit for each enrichment. Default value is 1000
#email.limit=
# Maximum attachment item limit for each enrichment. Default value is 100
#attachment.limit=
```

Credential: Name of the credential set you've just created on step 2. (i.e., Microsoft Exchange Credentials).

Trust Invalid SSL Certificates: Select this if certificate used on Exchange Server is self-signed or not recognized by browsers.

Require Approval From: Select user(s) from list to ask approval before executing actions on this integration.

Notify: Select user(s) from the list to notify when ATAR performs an action on this integration.

5. Click the Test button.

6. Click **Save** to complete integration.

Additional Notes

For Delete capability, at least one of the following parameters should be given:

- Email From

- Email Subject

- Email ID

- Attachment ID

And there are 3 deletion methods:

- **Hard Delete:** Deletes permanently (default)

- **Move To Trash:** Moves to trash folder (such as Deleted Items folder)

- **Soft Delete:** Moves to dumpster if it is enabled.

Integration Guide for Microsoft Windows DNS Server

Integration Overview

ArcSight SOAR uses Microsoft Windows DNS Server to redirect IP address to another IP address.

SOAR checks connection.secure parameter to connect via WinRM over http or https protocol.

Integration Capabilities

- Action
- Block

Configuration

Configuration on Microsoft Windows DNS Server

- SOAR connects to Microsoft Windows DNS Server's integration API via WinRM services. Therefore SOAR should be able to connect this service.
- WinRM credential is required.

Configuring ATAR

1. While creating this integration via Integrations tab of Configuration menu:

   Name: Display name of the integration.

   Type: Microsoft Windows DNS Server.

   Address: Address of the integration (the format should be http[s]://1.1.1.1:1234).

   Credential: WinRM credential is required. Credential that has been defined for this integration under the **Credentials** menu.

   Configuration: You need to specify the following configuration parameters.

   ```
   dns.zone.name: Redirected DNS server zone name
   dns.block.ip: Redirection address
   ```

```
dns.server.name: DNS server name
#Use https:// instead of http:// on WinRM connection
connection.secure=true : For secure connections, otherwise set to false.
#Parameters:
```

WindowsDNSCommandExecPath: Windows DNS command execution path.

Trust Invalid SSL Certificates: Select this if Engine's certificate used for the service is self-signed or not recognized by browsers.

Require Approval From: Select user(s) from list to ask her/his approval before executing actions on this integration.

Notify: Select user(s) from the list to notify when ATAR performs an action on this integration.

2. Click the **Test** button.
3. Click **Save** to complete integration.

Integration Guide for Microsoft Windows Services (WinRM)

Integration Overview

Integration Capabilities

- Action
- None

Configuration

Configuration on Microsoft Windows Services

- SOAR connects to Microsoft Windows Service's integration API via WinRM services.
- Therefore SOAR should be able to connect this service.
- WinRM credential is required.

Configuring SOAR

1. While creating this integration via Integrations tab of Configuration menu:

   Name: Display name of the integration.

   Type: Microsft Windows Services.

   Address: Address of the integration (the format should be 1.1.1.1 or abc.example.com).

   Configuration: You need to specify the following configuration parameters.

   putfile.generateuuid =

   putfile.defaultfolder =

   connection.secure = true

   Credential: Credential that has been defined for this integration under the Credentials menu.

   Trust Invalid SSL Certificates: Select this if certificate used for the service is selfsigned or not recognized by browsers.

Require Approval From: Select user(s) from list to ask her/his approval before executing actions on this integration.

Notify: Select user(s) from the list to notify when SOAR performs an action on this integration.

2. Click the **Test** button.

3. Click **Save** to complete integration.

Integration Guide for Microsoft Graph Security

Integration Overview

Microsoft Graph Security is an intermediary service (or broker) that provides a single programmatic interface to connect multiple Microsoft Graph Security providers such as Azure Security Center, Microsoft Defender APT, Microsoft Cloud App. Security, etc. Microsoft Graph Security integration lets you to search and manage security alerts created by those providers. This integration supports Microsoft Graph API v1.0.

Integration Capabilities

- Assign Alert
- Get Alert by ID
- List Alerts
- List Alerts by Category
- List Alerts by Destination
- List Alerts by Provider
- List Alerts by Severity
- List Alerts by Source IP
- List Alerts by Status
- Update Alert Comment
- Update Alert Feedback
- Update Alert Status

Prerequisites

ArcSight SOAR connects to **"login.microsoft.com"** and **"graph.microsoft.com"** APIs through HTTPS. Access to these services is required

.

Configuration

Configuring Microsoft Azure

1. Login to **https://portal.azure.com** and navigate to **Azure Active Directory** service.

2. Create a new registration in **App Registrations** menu following values.

   > Note: If an application is defined for other integrations, skip steps 1-3 to use it.

Name	Supported Account Types	Redirect URI
ArcSight SOAR	Accounts in this organizational directory only (Default Directory only - Single tenant)	(Web) https://localhost/soar

3. Click **Add a certificate or secret link** and create a new client secret. Specify the description and expiry period as 24 months.

4. Note the created **Secret Key** value along with Client ID.

5. Navigate to **API Permissions** and add the following permissions from Microsoft Graph:

Permission Type	Permission	Description
Application	SecurityEvents, ReadWrite, All.	Read and update your organization's security events.

6. Click **Yes** to grant admin consent for Default Directory.

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**.

2. Specify the following parameter values in the **Credential Editor** form:

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, Azure AD Credential).		Client ID of the application (for example, ArcSight SOAR) that is registered on Azure Portal.	Secret Key

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of the integration.
Type	Microsoft Graph Security

Parameter	Value
Address	Address of the integration (https://graph.microsoft.com/v1.0/security).
Configuration	Specify the following configuration parameters:

	tenant.id	**Tenant ID on Microsoft Azure tenant.id = ff1f0000-c600-4500-0038-9d4000000000**
	proxy.id	**ID of the Proxy integration if you access Microsoft Graph Security through a web proxy device. For example, proxy.id = 12345** .

Parameter	Value
Credential	Credential that has been defined for this integration under the **Credentials** menu.
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Save** to save the integration definition.

6. Navigate to **Configuration**>**Customization Library** and edit **Microsoft Graph Security Advanced Action Script Default Template**.

7. Select the integration that you have added to **Integrations** menu.

8. Click **Save** to complete the integration.

9. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Assign Alert**
   Action capability for assigning security alert to a person on Azure Security Center.

   • Rollback: No

   • Duplicate Control: No

   The following table presents the assign alert action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Assign to	Person this alert to be assigned to.	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

2. **Get Alert by ID**
   Enrichment capability for querying & retrieving security alert details by alert ID.

   The following table presents the get alert ID enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Alert ID	Alert ID on Azure Security Center.	String	No	Yes

Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

Following image provides the **Human Readable Output**:

3. **List Alerts**
   Enrichment capability for getting list of security alerts created in given time range. Results are based on the alerts creation time, displayed in newest to oldest manner..

   The following table presents thelist alerts enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Time Range	Time range filter for query.	Time range. Relative: e.g. Last 5 days Absolute: For example, 2021-08-14 15:10 – 2021-08-14 15:32	N/A	Yes

Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

Following image provides the **Human Readable Output**:

4. **List Alerts by Category**
   Enrichment capability for getting list of security alerts of a certain category created in given time range. Results are based on the alerts creation time, displayed in newest to oldest manner.

   The following table presents the list alert by category enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Alert Category	Category name	String	No	Yes

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Time Range	Time range filter for query.	Time range. Relative: e.g. Last 5 days Absolute: For example, 2021-08-14 15:10 – 2021-08-14 15:32	N/A	Yes

Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

Following image provides the **Human Readable Output**:

Created Time	Title	Severity	Status	Id
2021-08-20T04:42:24.8891406Z	[SAMPLE ALERT] Digital currency mining related behavior detected	high	newAlert	2517728662971264282_52d66d86-089d-4efb-8ba0-a9f0524f39a6
2021-08-13T10:38:16.2749644Z	[SAMPLE ALERT] Digital currency mining related behavior detected	high	newAlert	2517734497457406600_8d18f7ca-98d6-42ad-9de2-7e500b40b865

5. **List Alerts by Destination**

Enrichment capability for getting list of security alerts with the specified destination field, created in given time range. Results are based on the alerts creation time, displayed in newest to oldest manner.

The following table presents the list alerts by destination enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Destination	Destination Address.	Host Network Address URL .	Yes	Yes
Time Range	Time range filter for query.	Time range. Relative: e.g. Last 5 days Absolute: For example, 2021-08-14 15:10 – 2021-08-14 15:32	N/A	Yes

Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

Following image provides the **Human Readable Output**:

6. **List Alerts by Provider**

   Enrichment capability for getting list of security alerts originated from the specified security provider, created in given time range. Results are based on the alerts creation time, displayed in newest to oldest manner.

   The following table presents the list alerts by provider enrichment capability:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Provider	One of the Microsoft Security Providers.	String Azure Active Directory Identity Protection Azure Advanced Threat Protection Azure Security Center Azure Sentinel Microsoft Cloud App Security Microsoft Defender Advanced Threat Protection	No	Yes
Time Range	Time range filter for query.	Time range. Relative: For example, Last 5 days Absolute: For example, 2021-08-14 15:10 – 2021-08-14 15:32	N/A	Yes

   Output:

   Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

   Human Readable Output:

Following image provides the **Human Readable Output**:

7. **List Alerts by Severity**

Enrichment capability for getting list of security alerts with the specified severity value, created in given time range. Results are based on the alerts creation time, displayed in newest to oldest manner.

The following table presents the list alerts by severity enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Alert Severity	Alert severity set by vendor/ provider.	String High Medium Low Informational Unknown	No	Yes
Time Range	Time range filter for query.	Time range. Relative: For example, Last 5 days Absolute: For example, 2021-08-14 15:10 – 2021-08-14 15:32	N/A	Yes

Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

Following image provides the **Human Readable Output**:

8. **List Alerts by Source IP**

Enrichment capability for getting list of security alerts with the specified source IP field, created in given time range.Results are based on the alerts creation time, displayed in newest to oldest manner.

The following table presents the list alerts by source IP enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Source IP	Source IP Address.	Network Address	Yes	Yes
Time Range	Time range filter for query.	Time range. Relative: For example, Last 5 days Absolute: For example, 2021-08-14 15:10 – 2021-08-14 15:32	N/A	Yes

Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

Following image provides the **Human Readable Output**:

9. **List Alerts by Status**

   Enrichment capability for getting list of security alerts with the specified status value, created in given time range. Results are based on the alerts creation time, displayed in newest to oldest manner.

   The following table presents the list alerts by source enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Alert Status	Alert lifecycle status (stage).	String NewAlert InProgess Resolved Unknown	No	Yes
Time Range	Time range filter for query.	Time range. Relative: For example, Last 5 days Absolute: For example, 2021-08-14 15:10 – 2021-08-14 15:32	N/A	Yes

Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

Following image provides the **Human Readable Output**:

10. **Update Alert Comment**

    Action capability for adding/updating comment feild of the security alert.

    - Rollback: No

    - Duplicate Control: No

The following table presents the update alert comments action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Alert ID	Alert ID on Azure Security Center.	String	No	Yes
Alert Comment	Comment to be added to security alert.	String Closed in IPC Closed in MCAS	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

11. **Update Alert Feedback**
    Action capability for adding/updating feedback feild of the security alert.

    • Rollback: No

    • Duplicate Control: No
      The following table presents the update alert feedback action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Alert ID	Alert ID on Azure Security Center.	String	No	Yes
Alert Feedback	Comment to be added to security alert.	String Benign Positive False Positive True Positive Unknown	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

12. **Update Alert Status**
    Action capability for updating status of the security alert.

- Rollback: No

- Duplicate Control: No

The following table presents the update alert status action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Alert ID	Alert ID on Azure Security Center.	String	No	Yes
Alert Status	Comment to be added to security alert.	String In Progress New Alert Resolved Unknown	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

Integration Guide for MISP

Integration Overview

The MISP threat sharing platform is a free and open source software helping information sharing of threat intelligence including cyber security indicators.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with MISP.

- File Reputation
- IP Reputation
- URL Reputation
- Get Event
- Add Attribute to Event
- Add Tag to Event
- Create Event
- Create Event with Attribute
- Remove Attribute from Event
- Remove Tag from Event

ArcSight SOAR integrates with MISP to gather, store threat information and can query to IoCs. The capabilities can either be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

• Access to tcp port 443 as SOAR connects to MISP using HTTPS

• An API key for SOAR to connect to MISP

> **Note:** To gather the API key for SOAR, navigate to **MISP Interface** > **Event Actions** > **Automation**.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.

2. Specify the following parameter values in the **Credential Editor** form:

   a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, MISP Credentials)
Username	Empty
Password	Empty
Private Key	API Key retrieved from the MISP

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration form**:

Parameter	Value
Name	Display name of MISP integration on ArcSight SOAR
Type	MISP
Address	Address of the cloud service, in the following format: https://<misp_environement_ip>
Credential	Name of the credential set created in the previous step(For example, MISP Credentials)

Parameter	Value
Trust Invalid SSL Certificates	Not Applicable
Require Approval From	Select users from the list who can provide approval before executing enrichments on the integration
Notify	Select users from the list to notify when SOAR performs an enrichment on the integration

5. Click **Save** to complete the integration.

6. Navigate to **Configuration** > **Customization Library**.

7. In the **Customization Editor**, Edit **MISP Advanced Action Script Default Script Template** and for the **Integrations** field select the integration you saved (for example, MISP Integration).
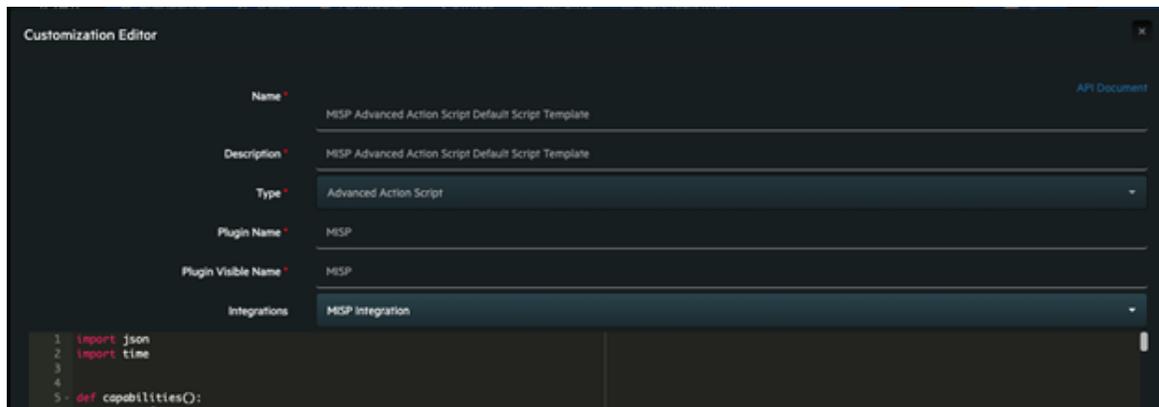
8. Navigate to **Configuration** > **Integrations**.

9. Click **Edit** for the MISP integration you created.

10. Click **Test** to test the integration.

Integration Guide for MxToolBox

Integration Overview

MxToolBox is a service that helps customers to make a query for domains and run the lookups.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with MxToolBox:

- Domain Blacklist Check
- Domain MX Check

Prerequisites

You must have access to HTTPS as the ArcSight SOAR connects to MxToolBox API through this service.

Configuration

Configuring MxToolBox

1. Login to MxToolBox and navigate to **Settings**, as shown in the following figure:
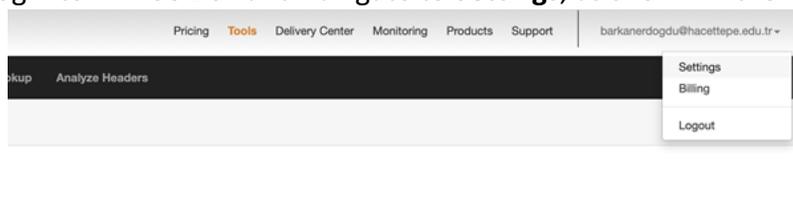
2. Click **Automation API Access Settings** in the Setting and add a new application.

3. Click **API Tab** and note the **API Key** to use on SOAR as shown in the following figure:
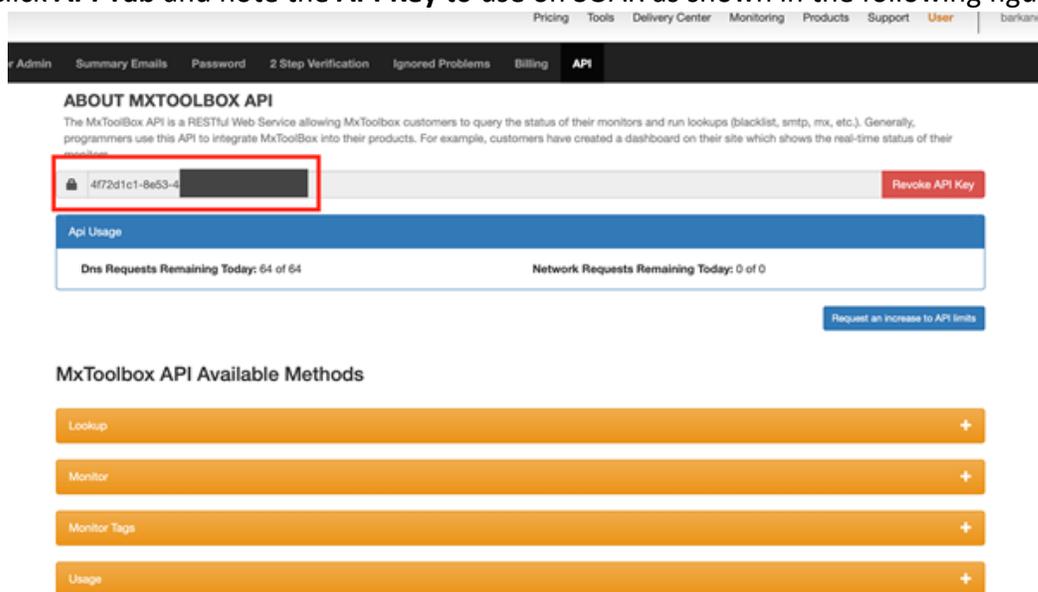
Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.

2. Specify the following parameter values in the **Credential Editor**

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, MxToolBox Credential).			API Key that is noted from the service

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration Form**.

Parameter	Value
Name	Display name of MxToolBox integration on SOAR.
Type	MxToolBox
Address	https://mxtoolbox.com
proxy.id	ID of the Proxy integration if you access mxtoolbox.com through a web proxy device. For Example: proxy.id = 12345.
Credential	Name of the credential set created on step 2(For example, MxToolBox Credentials).

Parameter	Value
Trust Invalid SSL Certificates	The SSL certificate of MxToolBox service is going to known by SOAR, so you do not need to check this box.
Required Approval From	Select users from the list who can provide approval before executing actions on this integration.
Notify	Select users from the list to notify when SOAR performs an action on this integration.

5. Click **Save** to save the integration definition.

6. Navigate to **Configuration** > **Customization Library** > **Open MxToolBox Script**

7. Select integration that is created at step 4 for **Integrations** field.

8. Click **Save** to complete the integration.

9. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Domain Blacklist Check**
   Enrichment capability for retrieving blacklist domain information.

   The following table provides the **Domain Blacklist Check** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Domain	Domain that you want to query.	Host	Yes	Yes
Integration	Name of the integration.	Integration	N/A	Yes

   Output:

   Case Scope: N/A

   Human Readable Output: Yes

2. **Domain MX Check**
   Enrichment capability for retrieving MX record information.

   The following table provides the **Domain MX Check** enrichment capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Domain	Domain that you want to query.	Host	Yes	Yes
Integration	Name of the integration.	Integration	N/A	Yes

   Output:

Case Scope: N/A

Human Readable Output:

Integration Guide for Okta

Integration Overview

Okta provides cloud software that helps organizations to manage and secure user authentication into applications.

This integration supports Okta API v1.0.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Okta:

- Activate User
- Add User Into Group
- Clear User Sessions
- Deactivate User
- Expire Password
- Get Group Members
- Get Groups For User
- Get User Details
- Get User Logs
- List Groups
- Remove User From Group
- Reset Password
- Set Temporary Password
- Suspend User
- Unlock User
- Unsuspend User

Prerequisites

- You must have access to HTTPS as the ArcSight SOAR connects to Okta API through this service.
- API Token is needed to access Okta REST API.

Configuration

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**.
2. Specify the following parameters in the **Credential Editor** form:

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, Okta API Credentials).			SSWS<space>API token

> **Note** : The API token must be specified in the Private Key field by adding SSWS term (SSWS<space><apitoken>).

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Specify the following parameter values in the **Configuration Form.**

Parameter	Value
Name	Display name of the integration.
Type	Okta
Address	Address of the integration (the format should be https://dev-1423.okta.com/).
Configuration	Specify the following configuration parameters: 	proxy.id	ID of the Proxy integration if you access Okta through a web proxy device. For example: proxy.id = 12345 .	
Credential	Credential that has been defined for this integration under the **Credentials** menu.
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Save** to save the integration definition.
6. Navigate to **Configuration**>**Customization Library** and edit **Okta Advanced Action Script Default Template**.
7. Select the integration that you have added to **Integrations** menu.
8. Click **Save** to complete the integration.

9. Click **Test**. **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Activate User**
   Action capability for activating an inactive user account.

   - Rollback: Yes

   - Duplicate Control: Yes

   The following table presents the **Activate User** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No

   Output:

   Case Scope: N/A

   Human Readable Output: N/A

2. **Add User into Group**
   Action capability for adding user account into a group.

   - Rollback: Yes

   - Duplicate Control: Yes

   The following table presents the **Add User into Group** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
Username	Username to be added to group.	Username Email Address Keyword Unknown	Yes	Yes
Group	Name of the group.	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

3. **Clear User Session**

Action capability for clearing user's active sessions.

- Rollback: No

- Duplicate Control: No

The following table presents the **Clear User Session** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
Username	User whose sessions will be cleared.	Username Email Address Keyword Unknown	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

4. **Deactivate User**

Action capability for deactivating a user account.

- Rollback: Yes

- Duplicate Control: Yes

The following table presents the **Deactivate User** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Username	Username to be deactivated.	Username Email Address Keyword Unknown	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

5. **Expire Password**
   Action capability for expiring user's password.

   - Rollback: No

   - Duplicate Control: No

The following table presents the **Expire Password** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
Username	User whose password will be set expired.	Username Email Address Keyword Unknown	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

6. **Get Groups Members**
   Enrichment capability for retrieving list of users belong to a group.

   The following table presents the **Get Groups Members** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Group	Group name to be queried for group memberships.	String	No	Yes

Output:

Case Scope

Action	Type	Category/ Value
Add	Scope Item	Username [Related]

Human Readable Output:

7. **Get Groups for User**

8. Enrichment capability for retrieving list of groups which user is member of..
   The following table presents the **Get Groups for User** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	User to be queried for group memberships.	Username Email Address Keyword Unknown	Yes	Yes

Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

9. **Get User Details**

Enrichment capability for retrieving user account details.

The following table presents the **Get User Details** action capability details:

Output:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	User to be queried.	Username Email Address Keyword Unknown	Yes	Yes

Case Scope

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

10. **Get User Logs**

Enrichment capability for retrieving log records for user account.

The following table presents the **Get User Logs** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
User	User to be queried.	Username Email Address Keyword Unknown	Yes	Yes

Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

11. **List Groups**

Enrichment capability for retrieving list of user groups.

The following table presents the **List Groups** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes

Output:

Case Scope:

Action	Type	Category/ Value
None	N/A	N/A

Human Readable Output:

12. **Remove User from Group**
    Action capability for removing user account from a group.

    - Rollback: Yes

    - Duplicate Control: Yes

    The following table presents the **Remove User from Group** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
User	User to be queried.	Username Email Address Keyword Unknown	Yes	Yes
Group	Name of the group	String	No	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

13. **Reset Password**
    Action capability for resetting user's password. A password reset mail is sent to user by Okta.

    - Rollback: No

    - Duplicate Control: No

    The following table presents the **Reset Password** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
Username	User whose password will be reset	Username Email Address Keyword Unknown	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

14. **Set Temporary Password**

Action capability for expiring user's password. This capabiliry sets **tempPassword** value of the user as **true**.

- Rollback: No

- Duplicate Control: No

The following table presents the **Set Temporary Password** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
Username	User whose password will be expired and "tempPassword" will be set to "True".	Username Email Address Keyword Unknown	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

15. **Suspend User**

Action capability for suspending user account**.**

- Rollback: Yes
- Duplicate Control: Yes

The following table presents the **Suspend User** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
Username	User account to be suspended.	Username Email Address Keyword Unknown	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

16. **Unlock User**
    Action capability for unlocking user account.

    - Rollback: No
    - Duplicate Control: No

The following table presents the **Unlock User** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
Username	User account to be unlocked.	Username Email Address Keyword Unknown	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

17. **Unsuspended User**

Action capability for unsuspending user account.

- Rollback: Yes

- Duplicate Control: Yes

The following table presents the **Unlock User** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
Username	User account to be suspended.	Username Email Address Keyword Unknown	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

Integration Guide for Ones BioAffix

Integration Overview

Ones BioAffix is a biometric single sign on (Biometric SSO) and biometric identity verification solution which lets organizations to manage their physical security and access.

Integration Capabilities

ArcSight SOAR has the following integration capability with Ones BioAffix:

- Change User Status (Block & Unblock)
- User Details (Info & Logs)

Use Case: Blocking Suspicious Employees

Integrated with Ones BioAffix ATAR lets users to investigate suspicious employee traffic through building and block access if needed. This can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Currently SOAR supports Ones BioAffix version 4.20.10.1.
- SOAR connects to Ones BioAffix API via HTTPS. Typically it runs on 8443/tcp* port. So access to this service is required.
- Credentials of administrator is required for SOAR to connect Ones BioAffix.

Configuration on Ones BioAffix

- No specific configuration is needed on Ones BioAffix server.

Configuring SOAR

1. Navigate **Configuration** > **Credentials** and click **Create Credential**.
2. Fill the Credential Editor form as follows:

   a. Internal Credential:

Type: Internal credential.

Name: Display name of credential set (i.e., Ones BioAffix Credentials)

Username: Administrator username you have on Ones BioAffix.

Password: Password for the administrator user you have on Ones BioAffix.

Private Key: Empty.

b. Credential Store:

Type: External credential.

Name: Name of the credential with pull path of the safe on store.

3. Navigate to **Configuration** > **Integrations** and click **Create Integration**.

4. Fill the configuration form as follows:

    Name: Display name of Ones BioAffix integration on ATAR.

    Type: Ones BioAffix Server.

    Address: Address of the integration (the format should be https://192.168.12.77:8443).

    Credential: Name of the credential set you've just created on step 2. (i.e., Ones BioAffix Credentials).

    Trust Invalid SSL Certificates: Select this if Engine's certificate is self-signed or not recognized by browsers.

    Require Approval From: Select user(s) from list to ask her/his approval before executing actions on this integration.

    Notify: Select user(s) from the list to notify when ATAR performs an action on this integration.

5. Click the **Test** button.

6. Click **Save** to complete integration.

Additional Notes

Due to API behaviour of Ones BioAffix integration, "Date of Birth", "Phone" and "Profile Photo" of users should be set to execute actions.

Integration Guide for Palo Alto Networks AutoFocus

Integration Overview

Palo Alto Networks AutoFocus is a threat intelligence platform which allows to search attack indicators and access to details of them. AutoFocus provides the intelligence, analytics, and context required to understand which attacks require immediate response and take decisive action to prevent future attacks.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Palo Alto Networks AutoFocus:

- Search Email Address
- Search File Hash
- Search File Name
- Search IP Address
- Search URL

Use Case: Investigating Phishing Campaigns

SOAR integrates with Palo Alto Networks AutoFocus to search attack indicators. SOAR can follow email inboxes for user's phishing reports and automatically creates an incident record on its service desk. During the investigation of the attack SOAR can extract the sender address, IP address, files in the attachment and ask these indicators to Palo Alto Networks AutoFocus if this is a known attack and previously analyzed. This can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- SOAR connects to Palo Alto Networks AutoFocus API via HTTPS. Access (https://autofocus.paloaltonetworks.com (443/tcp port) is required.
- An API key is required for SOAR to connect Palo Alto Networks AutoFocus.

Configuration on Palo Alto Networks AutoFocus

No specific configuration is needed. Login to https://autofocus.paloaltonetworks.com and note the API key under **Settings** > **General** menu.

Configuring SOAR

1. Navigate **Configuration** > **Credentials** and click **Create Credential**.
2. Fill the Credential Editor form as follows:

   a. Internal Credential:

   Type: Internal credential.

   Name: Display name of credential set (i.e., PAN AutoFocus Credential).

   Username: Empty.

   Password: API Key.

   Private Key: Empty.

   b. Credential Store:

   Type: External Credential.

   Name: Name of the credential with pull path of the safe on store.

3. Navigate **Configuration** > **Integrations** and click **Create Integration**.
4. Fill the configuration form as follows:

   Name: Display name of Palo Alto Networks AutoFocus integration on SOAR.

   Type: Palo Alto Networks AutoFocus.

   Address: Address of the integration (https://autofocus.paloaltonetworks.com).

   Credential: Name of the credential set you've just created on step 2. (i.e., PAN AutoFocus Credential).

   Configuration: You need to specify the following configuration parameters

   ```
   # Integration ID of the proxy integration to use when connecting to
   # current integration.
   # If not provided, SOAR will try to use a direct connection.
   #proxy.id=123
   # configure how far (in minutes) into the past this enrichment will look.
   # cache.reusing.duration=20
   ```

   Trust Invalid SSL Certificates: Select this if Engine's certificate is self-signed or not recognized by browsers.

Require Approval From: Select user(s) from list to ask approval before executing actions on this integration.

Notify: Select user(s) from the list to notify when SOAR performs an action on this integration.

5. The **EnrichmentFixedDelay** configuration parameter value must be set to less than 120 seconds because of AutoFocus' requirement. Otherwise AutoFocus API cookie will be expired.

6. Click the **Test** button.

7. Click **Save** to complete integration.

Integration Guide for Palo Alto Networks Panorama

Integration Overview

The Panorama management server provides centralized monitoring and management of multiple Palo Alto Networks next-generation firewalls and of WildFire appliances and appliance clusters.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Palo Alto Networks Panorama:

- Block IP address
- Block Host
- Block URL

Use Case: Blocking malicious IP addresses on multiple firewall appliances

With this integration, SOAR can block malicious IP addresses, hosts and URL addresses on multiple firewall devices simultaneously while responding cyber-attacks. This can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Currently SOAR supports Palo Alto Network Panorama 8.1.0 version.
- SOAR connects to Palo Alto Networks Panorama API using HTTPS. Access to 443/tcp port is required.
- An API key is required for SOAR to connect Palo Alto Networks Panorama.
- If users want to use multiple devicegroup, they should write devicegroup names comma separated, for ex: Ankara, Istanbul, Izmir

Configuration on Palo Alto Networks Panorama

1. Navigate to **Panorama** menu and create a new Admin Role for SOAR. The new role should be

   restricted to only specific XML API operations. Only required permissions are: "Configuration", "Operational Requests" and "Commit". Do not forget to disable all Web UI and Command Line permissions since they are unnecessary.

2. Create an Administrator account with **Custom Panorama Admin** type and SOAR API Role you have created in first step.

3. Commit all changes.

4. In order to obtain API key run the following request from command line.

```
curl -k -X GET 'https://Panorama_IP/api/?type=keygen& \
user=atarapi&password=password'
```

Configuration on SOAR

1. Navigate **Configuration** > **Credentials** and click **Create Credential**.

2. Fill the Credential Editor form as follows:

   a. Internal Credential:

   Type: Internal credential.

   Name: Display name of credential set (i.e., PAN Panorama Credential).

   Username: Empty.

   Password: Empty.

   Private Key: API Key you have created for SOAR.

   b. Credential Store:

   Type: External credential.

   Name: Name of the credential with pull path of the safe on store.

3. Navigate **Configuration** > **Integrations** and click **Create Integration**.

4. Fill the configuration form as follows:

   Name: Display name of Palo Alto Networks Panorama integration on SOAR.

   Type: Palo Alto Networks Panorama.

   Address: Address of the integration (https://10.0.2.254).

   Credential: Name of the credential set you've just created on step 2. (i.e., PAN Panorama Credential).

Trust Invalid SSL Certificates: Select this if Engine's certificate is self-signed or not recognized by browsers.

Configuration: You need to specify the following configuration parameters.

```
# Device group to use when adding and address object.
# This device group should be created in Palo Alto device before use.
# If users want to use multiple devicegroups, they should write
devicegroup
# names comma separated, for ex: Ankara, Istanbul, Izmir
devicegroup.name=HeadQuarters
# Address group to use when blocking IP addresses.
# This address group should be created in Palo Alto device before use.
addressgroup.ip=ATAR_BLOCK_IP
# Address group to use when blocking host names.
# This address group should be created in Palo Alto device before use.
addressgroup.host=ATAR_BLOCK_HOST
# Custom URL category to use when blocking URLs.
# This custom URL category should be created in Palo Alto device before
use.
custom.url.category=ATAR_BLOCK_URL
```

Require Approval From: Select user(s) from list to ask her/his approval before executing actions on this integration.

Notify: Select user(s) from the list to notify when SOAR performs an action on thisintegration.

5. Click the **Test** button.
6. Click **Save** to complete integration.

Integration Guide for Recorded Future

Integration Overview

Recorded Future is a threat intelligence service which collects and analyzes vast amounts of data to deliver relevant cyber threat insights in real time.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Recorded Future:

- Lookup Domain
- Lookup Hash
- Lookup IP Address
- Lookup URL
- Lookup Vulnerability
- Search Entity Lists
- Search Malware

Use Case: Investigating Phishing Campaigns

SOAR is integrated with Recorded Future, to help investigation and mitigation of phishing campaigns. When a phishing report email comes from user, SOAR extracts the indicators such as IP address, URLs and attachments in message and a new incident is created on SOAR's own Incident Management Service Desk. SOAR then asks these indicators to Recorded Future if this is a known attack and previously analyzed. This can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- SOAR connects to Recorded Future API via HTTPS. Access to https://api.recordedfuture.com/v2/ (443/tcp port) is required.
- An API key is required for SOAR to connect Recorded Future service.

Configuration on Recorded Future

Login to https://api.recordedfuture.com/v2/ and create a new API key under user Settings > API Access menu and note the API Key and API Password generated. This token is required by SOAR to access the platform for queries.

Configuring SOAR

1. Navigate **Configuration** > **Credentials** and click **Create Credential.**
2. Fill the Credential Editor form as follows:

   a. Internal Credential:

   Type: Internal credential.

   Name: Display name of credential set (i.e., Recorded Future Credentials).

   Username: API Key you have created on Recorded Future.

   Password: API Password for the key you have created on Recorded Future.

   Private Key: Empty.

   b. Credential Store:

   Type: External credential.

   Name: Name of the credential with full path of the safe on store.

3. **Navigate Configuration > Integrations and click Create Integration.**
4. Fill the configuration form as follows:

   Name: Display name of Recorded Future integration on SOAR.

   Type: Recorded Future.

   Address: Address of the integration (https://api.recordedfuture.com/v2/).

   Configuration: You need to specify the following configuration parameters.

```
# Integration ID of the proxy integration to use when connecting to
# current integration.
# If not provided, SOAR will try to use a direct connection.
#proxy.id=123
# configure how far (in minutes) into the past this enrichment will look.
#cache.reusing.duration=20
```

   Credential: Name of the credential set you've just created on step 2. (i.e., Recorded Future Credentials)

   Trust Invalid SSL Certificates: No need to select.

Require Approval From: Select user(s) from list to ask her/his approval before executing actions on this integration. Since SOAR only executes enrichments on Recorded Future, leave it empty.

Notify: Select user(s) from the list to notify when SOAR performs an action on this integration. Since SOAR only executes enrichments on Recorded Future, leave itempty.

5. Click on the **Test** button.

6. Click **Save** to complete integration.

Integration Guide for Robtex Lookup

1. Integration Overview

Robtex is used for various kinds of research of IP numbers, domain names, etc.

Robtex uses various sources to gather public information about IP numbers, domain names,

host names, Autonomous systems,routes, etc. It indexes the data in a big database and provide

free access for the data

2. Integration Capabilities

Action

Lookup

Configuration

Configuration on Robtex Lookup

SOAR connects to Robtex Lookup integrations via HTTPS. Therefore ATAR should be able to connect this service.

Configuring SOAR

1. While creating this integration via Integrations tab of Configuration menu:

   Name: Display name of Robtex lookup integration on SOAR.

   Type: Robtex lookup.

   Address: Address of the integration (the address should be https://www.robtex.com).

   Configuration: You need to specify the following configuration parameters

   ```
   # Integration ID of the proxy integration to use when connecting to
   # current integration.
   # If not provided, ATAR will try to use a direct connection.
   #proxy.id=123
   # configure how far (in minutes) into the past this enrichment will look.
   #cache.reusing.duration=20
   ```

Credential: Name of the credential set.

Trust Invalid SSL Certificates: Select this if Engine's certificate is self-signed or not recognized by browsers.

Require Approval From: Select user(s) from list to ask her/his approval before executing actions on this integration.

Notify: Select user(s) from the list to notify when ATAR performs an action on this integration.

2. Click the **Test** button.

3. Click **Save** to complete integration.

Integration Guide for Roksit DNS Firewall

Integration Overview

Roksit DNS Firewall is cloud-based cybersecurity service which provides web security and application control by analyzing DNS traffic.

Integration Capabilities

ArcSight SOAR has the following integration capability with Roksit DNS Firewall:

- Block hostname

Use Case: Blocking malicious hosts on DNS

With this integration, SOAR can block malicious hostnames on Roksit DNS Firewall service while responding cyber-attacks. This can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- SOAR connects to Roksit DNS Firewall API via HTTPS. So access to https://api.roksit.com (443/tcp port) is required.
- An API key is required to be created for SOAR to connect to Roksit DNS Firewall. Please contact to service provider.

Configuration on Roksit DNS Firewall

- No further configuration is needed.

Configuring SOAR

1. Navigate to **Configuration** > **Credentials** and click **Create Credential.**
2. Fill the Credential Editor form as follows:

   a. Internal Credential:

   Type: Internal credential.

Name: Display name of credential set (i.e., Roksit DNS FW Credentials).

Username: Empty.

Password: API Key you have obtained from Roksit.

Private Key: Empty.

b. Credential Store:

Type: External credential.

Name: Name of the credential with pull path of the safe on store.

3. **Navigate to Configuration > Integrations and click Create Integration.**

4. Fill the configuration form as follows:

   Name: Display name of Roksit DNS Firewall integration on SOAR

   Type: Roksit DNS Firewall

   Address: Address of the integration (address should be https://api.roksit.com).

   Credential: Name of the credential set you've just created on step 2. (i.e., Roksit DNS FW Credentials)

   Trust Invalid SSL Certificates: Select this if Engine's certificate is self-signed or not recognized by browsers.

   Require Approval From: Select user(s) from list to ask her/his approval before executing actions on this integration.

   Notify: Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Save** to complete integration.

6. Click **Test** to test the integration.

Additional Notes

- Roksit DNS Firewall integration on SOAR is defined as Advanced Action Script and content of the default script is accessible under **Configuration** > **Customization Library**.

- While defining the integration first time, you get a warning message as follows. For this type of integration this is the expected behaviour.

Integration Guide for RSA Security Analytics

Integration Overview

RSA Security Analytics provides real-time visibility into network traffic with full packet capture—on premises, in the cloud and across virtual infrastructure. It helps to detect threats as they traverse in the network, monitor the timing and movement of attackers across the network and reconstruct entire network sessions to support forensic investigations.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with RSA Security Analytics:

- Network Packet Capture (Time range)
- Network Packet Capture (Relative time)

Use Case: Investigating suspicious cases using packet captures

SOAR integrates with RSA Security Analytics to collect full packet capture for a given timeframe.During the investigation of an incident, SOAR can gather packet-capture from RSA Security Analytics with specified parameters such as offender IP, affected usernames, suspicious end-user machines, etc and put the related pcap file into incident timeline for further analysis and keeping evidence purposes. Collecting pcap files can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Currently SOAR supports RSA Security Analytics version 11.0.0.0.
- SOAR connects to RSA Security Analytics Network Concentrator's API via HTTP/HTTPS.
- By default API interface works on 50105/tcp port. So access permission to this port is required.
- A user account is required to be created for SOAR to connect to RSA Security Analytics Network Concentrator API.

Configuration on RSA Security Analytics Suite

1. Login to Security Analytics Suite and navigate to **ADMIN** > **Services** and then select **Concentrator** service and open up **Security** View by clicking **Actions** icon.

2. Add a new Role to be used for SOAR user. New role should have at least "sdk.content","sdk.manage" and "sdk.meta" permissions".

3. Add a new user with the role you have created in previous step.

Configuring SOAR

1. Navigate **Configuration** > **Credentials** and click **Create Credential**.

2. Fill the Credential Editor form as follows:

   a. Internal Credential:

   Type: Internal credential.

   Name: Display name of credential set (i.e., RSA Security Analytics Credential).

   Username: Username you have created for SOAR on RSA Security Analytics Suite.

   Password: Password of the user you have created for SOAR on RSA Security Analytics Suite.

   Private Key: Empty.

   b. Credential Store:

   Type: External credential.

   Name: Name of the credential with pull path of the safe on store.

3. Navigate **Configuration** > **Integrations** and click **Create Integration**.

4. Fill the configuration form as follows:

   Name: Display name of RSA Security Analytics integration on SOAR.

   Type: RSA Security.

   Address: Address of the integration (the format should be http[s]://192.168.1.10:50105 or http[s]://abc.example.com:50105).

   Credential: Name of the credential set you've just created on step 2. (i.e., RSA Security Analytics Credential)

   Trust Invalid SSL Certificates: Select this if device's certificate is self-signed or not recognized by browsers.

   Require Approval From: Select user(s) from list to ask her/his approval before executing actions on this integration.

Notify: Select user(s) from the list to notify when SOAR performs an action on thisintegration.

5. Click the **Test** button.

6. Click **Save** to complete integration.

Integration Guide for ServiceNow

Integration Overview

ServiceNow allows you to manage digital workflows for enterprise operations.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with ServiceNow:

- Close Incident
- Create Incident
- Update Incident

Prerequisites

You must have access to HTTPS as the ArcSight SOAR connects to ServiceNow API through this service.

Configuration

Configuring ServiceNow

1. Create a REST client on SOAR

    a. Login to SOAR platform.

    b. Navigate **Configurations > REST CLIENTS**.

    c. Create a new **REST client** by providing a description.

    > You must take a note of your **Client ID** and **Client Secret** as they would be used as username and password during configuring authentication later.

2. **Create a User**

    a. Login to **ServiceNow** platform.

    b. Navigate to **User Administration** > **User**.

    c. Click **New** to create a new user and specify the required credentials.Note: This username and password is used during the ArcSight SOAR configuration.

    d. Navigate to **User Administration** > **User** and edit the user you created newly to assign an admin role.

3. **Create Rest Messages**

    a. Navigate to **System Web Services > Outbound > REST Message**.

    b. Click **New** to create message and specify the following details in the form:

Form Fields	Values
Name	SOAR REST API Requests
Description	SOAR REST API Requests
Endpoint	<itom_host_url>/soar-api/api/v1
Authentication tab	
Authentication type	Basic
OAuth Profile	SOAR default_profile

    c. Click **Submit**.

    d. Navigate to **System Web Services** > **Outbound** > **REST Message** and select **SOAR REST API Requests**.

e. Click **New** and create following HTTP Methods within REST Messages:

i. **Update Case on SOAR Method**

Form Fields	Values
Rest Message	SOAR REST API Requests
Name	Update Case on SOAR
HTTP Method	Patch
Endpoint	<itom_host_url>/soar-api/api/v1/case/${serialId}
Authentication Tab	
Authentication type	Basic
OAuth Profile	SOAR default_profile
HTTP Request Tab	
HTTP Headers Section	
Name	**Value**
Content-Type	application/json
HTTP Query Parameters Section	
Content	${changes}

Specify the following details and click **Submit**:

ii. **Add Comment to SOAR Case**

Specify the following details and click **Submit**:

Form Fields	Values
Rest Message	SOAR REST API Requests
Name	Add Comment to the SOAR Case
HTTP Method	Post
Endpoint	<itom_host_url>/soar-api/api/v1/case-comment
Authentication Tab	
Authentication type	Basic
OAuth Profile	SOAR default_profile
HTTP Request Tab	
HTTP Headers Section	
Name	**Value**

Form Fields	Values
Content-Type	application/json
HTTP Query Parameters Section	

Content	`{`
	`"serialid":$(serialid),`
	`"comment":"$(comment)"`
	`}`

4. **Create Event Registry**

   a. Navigate to **Performance Analytics** > **System** > **Event Registry**.

   b. Click **New** to create an event registry and specify the following details in the form:

Form Fields	Values
Event Name	state_change_soar
Table	Incident[incident]

5. **Create Script Action**

   a. Navigate to **Browse System Policy** > **Events** > **Script Actions**.

   b. Click **New** to create script action and specify the following details in the form:

Form Fields	Values
Name	Update Case on SOAR
Event name	state_change_soar
Application	Global
Active	<Mark this checkbox>

   Add the following script:

```
try {
r = new sn_ws.RESTMessageV2('SOAR REST API Requests',
'Update Case on SOAR');
updated_fields = JSON.parse(event.parm2);
var serialId = updated_fields["serialId"];
if (updated_fields["caseProperties"] != {}) {
r.setStringParameterNoEscape('changes', JSON.stringify
(updated_fields["caseProperties"]));
r.setStringParameterNoEscape('serialId', serialId);
response = r.execute();
responseBody = response.getBody();
httpStatus = response.getStatusCode();
}
if (updated_fields["caseComment"] != {}) {
r = new sn_ws.RESTMessageV2('SOAR REST API Requests',
```

```
'Add Comment to SOAR Case');
r.setStringParameterNoEscape('serialId', serialId);
var comment = updated_fields["caseComment"]["comment"]
["value"];
r.setStringParameter('comment', comment.replace(/\n/g,
 " "));
response = r.execute();
responseBody = response.getBody();
httpStatus = response.getStatusCode();
}

} catch (ex) {
var message = ex.message;
}
```

6. **Create Business Rules**

    a. Navigate to **System Definition** > **Business Rules**.

    b. Click **New** to create business rule and specify the following details in the form:

Form Fields	Values
Name	soar-rule
Table	Incident[incident]
Application	Global
Active	<Mark this checkbox>
Advanced	<Mark this checkbox>
When to run tab	
When	after
Order	1

Form Fields	Values
Update	<Mark this checkbox>
Advanced tab	
Script	Add the following script:

```
if ((current.operation() == 'update' && current.state.changes() ||
current.description.changes()) || current.comments.changes()) {
var currentValues = {
"caseProperties":{},
"caseComment":{},
"serialId": current.short_description.toString().split("-")[0]
};
var previousValues = {
"state": previous.state.getDisplayValue(),
"description": previous.description.getDisplayValue(),
"comments": previous.comments.getJournalEntry(1)
};
if (current.comments.changes()){
currentValues["caseComment"]["comment"] = {"value":
current.comments.getJournalEntry(1)};
}
if (current.state.changes()){
currentValues["caseProperties"]["status"] = {"value":
current.state.getDisplayValue()};
}
if (current.description.changes()){
currentValues["caseProperties"]["description"] = {"value":
current.description.getDisplayValue()};
}
gs.eventQueue('state_change_soar', current, JSON.stringify(previousValues),
JSON.stringify(currentValues));
}
```

    c. Click **Submit**.

7. **Import Certificate** (if SOAR has self-signed certificate)

    a. Navigate to System **Definition** > **Certificates**.

    b. Click **New** to create new certificate entry.

    c. Click the attachment icon below to **upload your certificate file**. Run the following command to create the certificate

```
openssl s_client -connectt cdfhost:cdfport 2>/dev/nul
?dev/null | sed -ne '/-BEGIN CERTIFICATE-/,/-END
CERTIFICATE-/p'
```

    d. Save the content with .der extension.

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**

2. Specify the following parameters in the **Credential Editor** form:

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, ServiceNow Credentials).	Username of the created user on ServiceNow	Password of the created user on ServiceNow	Empty

   Check the Cleartext Access option.

3. Click **Configuration** > **Lists** > **Create Lists**. The list must two columns with the type **Keyword**. Specify a name for that list and save it. The name of the list is used during integration configuration.

4. Click **Configuration** > **Integrations** > **Create Integration**.

5. Specify the following parameter values in the **Configuration Form.**

Parameter	Value
Name	Display name of the integration.
Type	ServiceNow
Address	Address of the ServiceNow integration (the format should be https://dev107155.service-now.com).
Configuration	Specify the following configuration parameters: **proxy.id** — ID of the Proxy integration if you access ServiceNow through a web proxy device. For example: proxy.id = 12345 . **list.name** — List name that is used for mapping ArcSight SOAR cases to ServiceNow incidents. For example, list.name=servicenowMapList
Credential	Credential that has been defined for this integration under the **Credentials** menu.
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

6. Click **Save** to save the integration definition.

7. Navigate to **Configuration>Customization Library** and edit **ServiceNow Advanced Action Script Default Template**.

8. Select the integration that you have added to **Integrations** menu.

9. Click **Save** to complete the integration.
10. Click **Test**. **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Create Incident**
   Action capability for creating incident on ServiceNow.

   - Rollback: No

   - Duplicate Check: Yes

   The following table presents the **Create Incident** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Category and SubCategory	Category and Subcategory information of created incident	Enum	No	Yes
Description	Servicenow Incident Description	Text	No	Yes
Impact	ServiceNow incident impact	Enum	No	Yes
Urgency	Servicenow Incident Urgency	Enum	No	Yes
Comment	Servicenow Incident Comment	Text	No	Yes
Assignment Group	Servicenow Incident Assignee	Text	No	Yes

2. **Close Incident**
   Action capability for closing incident on ServiceNow.

   - Rollback: No

   - Duplicate Check: No

   The following table presents the **Close Incident** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
State	Closing State of Servicenow incident.	Enum	No	Yes
Resolution Code	Resolution Code for ServiceNow incident.	Enum	No	Yes

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Resolution Note	Resolution Notefor ServiceNow incident.	Enum	No	Yes

3. **Update Incident**

   Action capability for updating incident on ServiceNow.

   The following table presents the **Update Incident** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Description	Servicenow Incident Description	Text	No	No
Impact	ServiceNow incident impact	Enum	No	No
Urgency	Servicenow Incident Urgency	Enum	No	No
Comment	Servicenow Incident Comment	Text	No	No
Assignment Group	Servicenow Incident Assignee	Text	No	No
State	ServiceNow incident status	Enum	No	No

   Output:

   Case Scope: N/A

   Human Readable Output: Yes

Integration Guide for SMTP Mail Server

Integration Overview

ArcSight SOAR uses the SMTP Server to send emails and notification messages. ATAR can also use the same integration to access inboxes to read emails, such as device action approvals if it is configured as an IMAP server.

Integration Capabilities

- Action
- Send email

Configuration

Prerequisites

- SOAR connects to SMTP Mail Server integration via Simple Mail Transfer Protocol.Therefore SOAR must be able to connect this service.
- A user's credential is required for SMTP AUTH. The same credential will be used if IMAP is configured.

Configuring SOAR

1. Click **Configuration** > **Integrations** > **Create Integration**.
2. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of the SMTP Mail Server integration..
Type	SMTP Mail Server
Address	Address of the integration (the format should be 1.1.1.1 or abc.example.com).

Configuration	Specify the following configuration parameters:
	mail.default-encoding is the encoding format of emails.
	mail.transport.protocol is the default message transport protocol.
	mail.smtp.auth specifies whether SMTP Authentication will be enabled or not. It can
	be "true" or "false".
	mail.smtp.port is the port for the SMTP service.
	mail.smtp.starttls.enable specifies whether TLS for SMTP will be enabled or not. It
	can be "true" or "false".
	mail.store.protocol is the protocol to access inboxes (for email reading). Default
	value is "imaps".
	mail.imaps.host is the address of the IMAPS server.
	mail.imaps.port is the port for IMAPS service.
Credential	Credential that has been defined for this integration under the **Credential** menu.
Trust Invalid SSL Certificates	Select this if Engine's certificate certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

3. Click **Test** to test the integration.

4. Click **Save** to complete the integration.

Additional Notes

- If a SMTP integration is used without credentials then it can't be used as incoming e-mail processor and for approvals.

- The global configuration parameter EMailDevice, under the Parameters tab of **Configuration** menu, configures the default mail server to be used in sending notifications and emails. Therefore, you must set the value of this parameter to the ID value for the SMTP Mail.

Integration Guide for Sophos XG Firewall

Integration Overview

Sophos XG Firewall is an integrated security platform featuring next gen firewall capabilities.

Integration Capabilities

ArcSight SOAR has the following integration capability with Sophos XG Firewall:

- Block IP
- Block FQDN
- Block URL
- Block Email Sender

Use Case: Blocking bad actors on firewalls

With this integration, SOAR can block malicious IP addresses, hosts and URL addresses on firewall devices while responding cyber-attacks. This can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- SOAR connects to Sophos XG Firewall API via management port. So access permission
- to this port is required.
- A user account for SOAR to connect to Sophos XG Firewall.

Configuration on Sophos XG Firewall

1. Click **Configure** > **Authentication** > **Users menu** and add an administrator user account.
2. Create a new profile or select a suitable one from the Profile list. Profile should have the following permissions:
   - Read-write for Objects
   - Web & content filter

- Email protection
- None for the rest of the permissions

3. Navigate to **Backup & Firmware** > **API** to enable API Configuration and add SOAR IP Address to the Allowed IP Address list.

4. Click **Administration** > **Device Access** to ensure that SOAR's assigned zone can access the HTTPS service of Sophos. You can prefer to create a Local Service ACL Exception Rule as well. For more information consult the Sophos How to use API documentation for further information.

1. Click **Configuration** > **Credentials** > **Create Credential**.

2. Specify the following parameter values in the **Credential Editor**:

    a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Sophos XG Credentials)
Username	Username you have created on firewall.
Password	Password you have created on firewall.
Private Key	Empty

    b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Sophos XG integration on SOAR.
Type	Sophos XG Firewall.
Address	Address of the firewall (the format should be https://192.168.10.1:4444)

Parameter	Value
Configuration	Specify the following configuration parameters: ```# IP host group name for adding ip hosts to block
iphost.group.name=ATAR_IP_BLOCK
# FQDN host group name for adding fqdns to block
fqdnhost.group.name=ATAR_HOST_BLOCK
# Web filter url group name for adding urls to block
webfilterurl.group.name=ATAR_URL_BLOCK```
Credential	Name of the credential set created on step 2 (For example, Sophos XG Credentials)
Trust Invalid SSL Certificates	Select this if Management UI's certificate certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration. Since SOAR only executes enrichments on Symantec DLP, leave it empty

5. Click **Test** to test the integration.

6. Click **Save** to complete the integration.

Additional Notes

- IP, FQDN and URL filter groups are automatically created by SOAR if they don't exist.1. IP, Host and URL filter groups are automatically created by SOAR if they don't exist.

- Sophos XG Firewall URL Filtering only accepts URLS with the following format http://www.example.com. URI paths are not accepted through API. Therefore SOAR transparently trim the URI part while submitting to Sophos XG Firewall.

- SOAR stores blocked email addresses in a list to keep track. Sophos currently does not provide a method to get the current list and any update will overwrite the list with the new address so administrator should only update the MTA Blocked Sender List through SOAR. Also this list is kept for each different Sophos integration but creating a second integration for the same device can lead to data inconsistency.

Integration Guide for SORBS Query

Integration Overview

SORBS Query provides free access to its DNS-based Block List to effectively block mail from more than 12 million host servers known to disseminate spam, phishing, attacks and other forms of malicious emails.

Integration Capabilities

- Action
- Check IP

Configuration

Configuration on SORBS Query

- ATAR connects to SORBS integrations's API via HTTPS. Therefore ATAR should be able to connect this service.

Configuring SOAR

Configuring SOAR

1. Click **Configuration** > **Integrations** > **Create Integration**.
2. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of SORBS Query integration on SOAR.
Type	SORBS Query.
Address	Address of the integration (the address should be http[s]://dnsbl.sorbs.net).

Parameter	Value
Trust Invalid SSL Certificates	Select this if Engine's certificate certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Not Applicable

3. Click **Test** to test the integration.
4. Click **Save** to complete the integration.

foIntegration Guide for Symantec Advanced Threat Protection

Integration Overview

Symantec Advanced Threat Protection is Symantec's endpoint protection platform closely works with SEP Manager.

Integration Capabilities

- Action Capabilities
- Quarantine Endpoint (isolate_endpoint and rejoin_endpoint)
- Delete File From Endpoint (delete_endpoint_file)
- Enrichment Capabilities
- Get Events (/events)

Configuration

Configuring Symantec Advanced Threat Protection

Symantec ATP uses https (tcp/443) for API access by default.

1. Click **Settings** > **Data Sharing** > **OAuth Clients** > **Add application with custo role** to add the API application.
2. The image in the **Privileges** section represents how the custom role must be configured. After creating user, Symantec displays the **client secret** and **client id**, which is used in SOAR configuration modal.

Configuring SOAR

1. Navgate to **Configuration** > **Integrations**.
2. Specify the following parameter values in the **Integrations Editor**:

Parameter	Value
Name	Display name of Symantec Advanced Threat Protection integration on SOAR
Type	Symantec Advanced Threat Protection.
Address	Address of the integration (in the following format: https://1.1.1.1)
Configuration	Specify the following configuration parameters. `#EVENT_RESULT_LIMIT`
Credential	Name of the credential set created under the **Credentials** menu. You must use **client id** as username and **client secret** as password.
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

3. Click **Save** to save the integration.

4. Navigate to **Configuration**>**Customization Library** and edit **Symantec Advanced Threat Protection Advanced Action Script Default Script Template**.

5. Select the integration that you have added to Integrations menu.

6. Click **Save** to complete the integration.

7. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

Integration Guide for Symantec Bluecoat Malware Analysis Appliance (MAA)

Integration Overview

Symantec Bluecoat MAA is a malware analyzer sand-box solution. SOAR uses Symantec Bluecoat Malware Analysis Appliance to analyze files and URLs.

Integration Capabilities

- Action
- File Analysis
- Hash Analysis
- URL Analysis

Prerequisites

- SOAR connects to Symantec Bluecoat MAA's Remote API (RAPI) via HTTPS. Therefore, SOAR should be able to connect this service.
- A user credential with admin role and its token are required.

Configuration

Configuring Symantec Bluecoat Malware Analysis Appliance (MAA)

- To generate a token, click **System Settings** > **Users** > **Add New API Key** on the appliance.

Configuring SOAR

1. Navgate to **Configuration** > **Integrations**.
2. Specify the following parameter values in the **Integrations Editor**:

Parameter	Value
Name	Display name of Symantec Bluecoat MAA integration on SOAR.
Type	Symantec Bluecoat MAA .
Address	Address of the integration (in the following format: http[s]://1.1.1.1:1234
Credential	Name of the credential set created under the **Credentials** menu. The user's API token should be set as the password while creating integration credential.
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

3. Click **Test** to test the integration.
4. Click **Save** to save the integration.

Integration Guide for Symantec BlueCoat Proxy SG

Integration Overview

BlueCoat Proxy SG is a secure web gateway solution developed by Symantec which controls the users' access to web content.

Integration Capabilities

SOAR has the following integration capability with Symantec BlueCoat Proxy SG

- Block

Use Case: Blocking access to malicious URL

SOAR can integrate with Symantec BlueCoat Proxy SG to block malicious URLs detected while responding an incident. Blocking can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Currently SOAR supports Symantec BlueCoat Proxy SG version 6.6.4.2 and connects to Symantec BlueCoat Proxy SG Management UI through HTTPS in order to download existing copy of local database. As Management Console runs on 8082 /tcp port, so access to this port is required.
- SOAR connects to Symantec BlueCoat Proxy SG via SSH to immediate update of local database. So access to 22/tcp port is required.
- Symantec BlueCoat Proxy SG connects back to SOAR API to gather new copy of the local database. As SOAR API runs on 443/tcp port, so access from BlueCoat Proxy SG to this service is required.
- Admin user credentials are required for SOAR to connect Symantec BlueCoat Proxy SG

Configuring Symantec BlueCoat Proxy SG

1. Click **Configuration** > **Content Filtering** > **General** and enable **Local Database**.

2. Click **Configuration** > **Content Filtering** > **Local Database** and configure copy of local database URL accessible on SOAR . The format should be `https://cdf/soar-api/api/bluecoat/list/integrationId}`

   `integrationId`: ID of BlueCoat Proxy SG integration on SOAR.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.

2. Specify the following parameter values in the **Credential Editor**:

   a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, BlueCoat Proxy SG Credentials)
Username	Username of the administrator
Password	Password of the admin user
Private Key	Empty

   b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Symantec BlueCoat Proxy SG integration on SOAR
Type	Symantec BlueCoat Proxy SG
Address	Address of the integration (in the following format: https://192.168.2.99)

Parameter	Value
Configuration	Specify the following configuration parameters:

	<pre># Default category to block URLs. If empty, value of
	# BlueCoatDefaultBlockListCategoryName configuration
	#category=soar
	# parameter will be used.
	# Comma (,) separated list of IP addresses of Bluecoat
	# servers that are allowed to retrieve blocked URL list.
	# servers that are allowed to retrieve blocked URL list.
	# servers that are allowed to retrieve blocked URL list.
	#allowedaddresses=
	# Default block list source URL. This URL should be pointed out
	# third-party block list source address. If unspecified, value
	# of BlueCoatDefaultBlockListURL will be used.
	#blocklistsource=
	# Connect to Bluecoat Proxy using SSH with provided
	# credential and execute commands to immediately force
	# refresh of the block list. Default is false.
	#forcerefresh.enabled=false</pre>

	For a third party blacklist to work correctly it must be structured as follows:
	For example,
	If you want to work with seperate categories you can give a different category name to differentiate between SOAR sourced URL's and the third-party URL's.

	<pre>define category "soar"
	www.example.com
	www.example.com/example.asp
	example.com
	192.168.201.57
	end category "soar"</pre>

Credential	Name of the credential set created on step 2 (For example, BlueCoat Proxy SG Credentials)
Trust Invalid SSL Certificates	Select this if Management Consoles's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Not Applicable

5. Click **Save** to complete the integration.
6. Click **Test** to test the integration.

Additional Notes

- Due to update mechanism of Blucoat Proxy SG's Content Filter/Local Database, BlueCoat Proxy SG retrieves the list of items to be blocked from a URL located on a web server that is

accessible by the Proxy SG. SOAR maintains a copy of Content Filter/Local Database and is accessible on `https://cdf/soar-api/api/bluecoat/list/integrationId}`.

- SOAR connects to management console and downloads a copy of the Content Filter/Local Database before adding new entries. If SOAR is the only place managing Content Filter/Local Database, you don't need to provide this access since SOAR always has the latest copy.

- After updating the list of items to be blocked on itself, SOAR might connect to BlueCoat Proxy SG via SSH and trigger an immediate download of the Content Filter/ Local Database file. This operation requires to access privileged-mode. In order to use this method set `forcerefresh.enabled=true` on integration configuration. List of commands executed during this operation can be found under **Configuration** > **Customization Library** > **Symantec BlueCoat Proxy SG SSH Integration Action** (Block) Default Template.

- If **Automatically check for updates** is set on Content Filter/Local Database configuration BlueCoat periodically connects and checks the latest version of the list. If you don't want immediate update you may set `forcerefresh.enabled=false` on integration configuration and prefer to use automatic updates.

Integration Guide for Symantec Bluecoat Site Review

Integration Overview

Bluecoat Site Review is a site to report uncategorized URLs to Symantec/Bluecoat.

Integration Capabilities

- Action
- Report Uncategorized URL (should get URL from scope)

Configuration

Configuration on Bluecoat Site Review

No requirements

Configuring SOAR

- In SOAR **Configuration**, specify **Name**, **Address** and **submissionEmailAddress** to check submission result from returning mail.

> Note: Add a dummy credential that can be removed in future releases.

Integration Guide for Symantec Data Loss Prevention (DLP)

Integration Overview

Symantec DLP is a solution to ensure that sensitive data is not lost, misused, or accessed by unauthorized users

Integration Capabilities

SOAR has the following integration capabilities with Symantec DLP:

- Retrieve incidents

Use Case: Investigating Suspicious Behaviour

During investigation of a suspicious behaviour of an employee or an endpoint, SOAR integrated with Symantec DLP, can get access the related DLP incidents for better understanding of the case. Investigation can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Currently SOAR supports Symantec DLP 14.6.0200 version. SOAR connects to Symantec DLP API via HTTPS. Access to 443/tcp port is required.
- A user account is required for SOAR to connect to Symantec DLP.

Configuring Symantec DLP

1. Login to Symantec DLP Enforce Server and navigate to **System** > **Login Management** > **Roles** to create a web service role. The web service role should have the following permissions:
   - Incidents: View
   - Perform Attribute Lookup
   - Incident Reporting and Update API: Incident Reporting

- Display Attributes: All,

- Custom Attributes: View all

2. Click **System** > **Login Management** > **DLP Users** and add a DLP user account with the role that is created on previous step.

3. Login to Symantec DLP Enforce server administration console with the DLP user account created in previous step.

4. Click **Incidents** > **Incident Reports** and select a system defined incident list, such as **Incidents - All**.

5. Edit report filters to narrow down the results to be returned if needed. In the **Summarize by** menu verify that **and** are both selected.

6. Save the report as a new private report and note the new report's ID.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.

2. Specify the following parameter values in the **Credential Editor**:

   a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Symantec DLP Credentials)
Username	User you have created for SOAR on Symantec DLP.
Password	Password of the user you have created for SOAR on Symantec DLP
Private Key	Empty

   b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Symantec DLP integration on SOAR.
Type	Symantec Data Loss Prevention.

Parameter	Value
Address	Address of the integration (in the following format: https://192.168.2.15)

Configuration	Specify the following configuration parameters:
	
	`# Report id`
	`report.id=221`

Credential	Name of the credential set created on step 2 (For example, Symantec DLP Credentials)
Trust Invalid SSL Certificates	Select this if Web UI's certificate certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration. Since SOAR only executes enrichments on Symantec DLP, leave it empty

5. Click **Test** to test the integration.
6. Click **Save** to complete the integration.

Additional Notes

For the details of web service role and report creation please refer to Symantec™ Data Loss Prevention Incident Reporting and Update API Developers Guide.

Integration Guide for Symantec DeepSight Intelligence

Integration Overview

Symantec DeepSight Intelligence is a commercial threat intelligence service which provides actionable intelligence with context and technical details surrounding a threat so teams can quickly assess cyber risk and implement proactive controls.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Symantec DeepSight Intelligence Service:

- Ingest intelligence data as alert
- Domain Query
- File Query
- IP Query

Use Case: Investigating Phishing Campaigns

SOAR is integrated with Symantec DeepSight Intelligence, to help investigation and mitigation of phishing campaigns. When a phishing report email comes from user, SOAR extracts the indicators such as IP address, domains and attachments in message and a new incident is created on SOAR's own Incident Management Service Desk. SOAR then asks these indicators to Symantec DeepSight Intelligence if this is a known attack and previously analyzed. This can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- SOAR connects to Symantec DeepSight API via HTTPS. Access to https://deepsightapi.symantec.com/v1/ (443/tcp port) and https://datafeeds.symantec.com/ (443/tcp port) is required.
- A user account and a certificate-password pair are required for SOAR to connect to Symantec DeepSight. These will be supplied by Symantec through DeepSight portal.

Configuring Symantec DeepSight Intelligence

SOAR requires a username and password to be created on Symantec DeepSight for authentication purposes for Alert Source. If enrichment capabilities are to be used an API key must be enabled and created.Use an administrator account to enable API Access for the account you wish to use in SOAR.

1. Select **user's detail** tab. The tab includes a section for DeepSight API Token. Select **Enable Access**
2. Login with the SOAR account to the DeepSight portal.
3. Click **Settings** > **My Profile** and locate the **DeepSight API Token** tab.
4. Copy the API key.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor**:

   a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Symantec DeepSight Credentials).
Username	Empty
Password	API Key you've get from Symantec DeepSight Intelligence platform.
Private Key	Empty

   b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

Configuring Symantec DeepSight Intelligence as Alert Source

1. Click **Configuration** > **Alert Source** > C**reate Alert Source Configuration**.
2. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Symantec DeepSight Intelligence Alert Source on SOAR.
Type	Symantec DeepSight Intelligence Datafeeds
Address	Address of the Symantec DeepSight Intelligence DataFeeds (https://datafeeds.symantec.com/v1/).

Configuration	Specify the following configuration parameters:
	
	``# Number of item to ingest per data feed type on first integration
	alertCountPerFeedType=1000
	# Minimum item reputation value to turn into Alert on SOAR
	minReputationToAlert=10
	#usable behaviour names :
	attack,attacks,bot,cnc,fraud,malware,phish,spam,phish_host
	#behaviourNames=attack,bot,CnC,fraud,malware,spam
	# Integration ID of the proxy integration to use when connecting to current source.
	# If not provided, SOAR will try to use a direct connection.
	#proxy.id=5422
	# configure how far (in minutes) into the past this enrichment will look.
	#cache.reusing.duration=20```

Credential	Name of the credential set created on step 2 (For example, Symantec DeepSight Credentials)
Trust Invalid SSL Certificates	Select this if Web UI's certificate certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration. Since SOAR only executes enrichment on Symantec DeepSight, leave it empty.
Visible Alert Fields	You may define which alarm fields will be displayed on Incident Management Service Desk.

Configuring Symentec DeepSight Intelligence as Integration

1. Click **Configuration** > **Alert Source** > C**reate Alert Source Configuration**.
2. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Symantec DeepSight Cyber Intelligence integration on SOAR.
Type	Symantec DeepSight Cyber Intelligence
Address	Address of Symantec DeepSight Cyber Intelligence (https://deepsightapi. symantec.com/v1)
Configuration	Specify the following configuration parameters: ```# Integration ID of the proxy integration to use when connecting to current integration. # If not provided, SOAR will try to use a direct connection. #proxy.id=123 # configure how far (in minutes) into the past this enrichment will look. #cache.reusing.duration=20```
Credential	Name of the credential set created on step 2 (For example, Symantec DeepSight Credentials)
Trust Invalid SSL Certificates	Select this if Web UI's certificate certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration. Since SOAR only executes enrichment on Symantec DeepSight, leave it empty.
Notify	Select users from the list to notify when SOAR performs an action on this integration. Since SOAR only executes enrichment on Symantec DeepSight, leave it empty.

3. Click **Test** to test the integration.

4. Click **Save** to complete the integration

Integration Guide for Symantec Endpoint Protection Manager

Integration Overview

Symantec Endpoint Protection Manager (SEP Manager) is a management platform for security software suite, which consists of anti-malware, intrusion prevention and firewall features for server and desktop computers.

Integration Capabilities

SOAR has the following integration capabilities with Symantec Endpoint Protection Manager:

- Start Scan on Client
- Block File Hash
- Get Client Info

Use Case: Starting scan jobs on suspicious endpoints.

During the course of and investigation or responding a ongoing cyber-attack, it is required to run scan jobs on suspicious endpoints to validate the threat. SOAR can start scan jobs on Symantec Endpoint Protection Manager to help on deciding the next course of action.

This can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Currently SOAR supports Symantec Endpoint Protection Manager 14.2.760 version. SOAR needs to connect Symantec Endpoint Protection Manager API and Database.
- Access to 8443/tcp, 8446/tcp port for API acceess and 1433/tcp, 1434/udp port for database access is required.
- User accounts for API access and database access are required for SOAR to connect to Symantec Endpoint Protection Manager.

Configuring Symantec Endpoint Protection Manager

1. Login to SEP Management Server on https://SEPManager:8443/console/apps/sepm and create an administrator account on **Admin** tab.

2. Click **Policy** > **Policy Components > File Fingerprint Lists** and add a File Fingerprint List.

3. You might create a file containing MD5 value of eicar.com test signature 44d88612fea8a8f36de82e1278abb02f: to upload a file to create the list.

4. Login to SEP Manager Web Service Application Registration on https://SEPManager:8443/sepm with the admin account you've created on previous step and register a webservice application to be used by SOAR.

   > **Note** the Client ID and Client Secret values are generated.

5. Create a database user that has selected permissions and ensure that the SQL Browser service is configured and running on MSSQL Server.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.

2. Specify the following parameter values in the **Credential Editor**:

   a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, SEP Manager Credentials).
Username	Username you have created for SOAR on Symantec Endpoint Protection Manager
Password	Password of the user you have created for ATAR on Symantec Endpoint Protection Manager.
Private Key	Empty

   b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

3. To create credentials to be used for database connection:

a. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, SEP Manager DB Credentials).
Username	Database username you have created for SOAR on SEP Manager Database.
Password	Password of the user you have created for SOAR on SEP Manager Database.
Private Key	Empty

b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

4. Click **Configuration** > **Integrations** > **Create Integration**.

5. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Symantec Endpoint Protection Manager integration on SOAR
Type	Symantec Endpoint Protection Manager
Address	Address of the integration (in the following format: https://192.168.2.140)
Configuration	Specify the following configuration parameters: ```client.id=xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx client.secret=xxxxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx #domainName= directdbaccess.enabled=true directdbaccess.jdbcurl= jdbc:sqlserver://192.168.2.140:1433\\SEPMDB;database=sem5 directdbaccess.credential=33323 # Integration ID of the proxy integration to use when connecting to # current integration. # If not provided, ATAR will try to use a direct connection. #proxy.id=123```
Credential	Name of the credential set created on step 2 (For example, SEP Manager Credentials).
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select user(s) from the list to notify when ATAR performs an action on this integration.

6. Click **Test** to test the integration.

7. Click **Save** to complete the integration.

Additional Notes

Symantec Endpoint Protection Manager Webservice registration works on 8446/tcp port by default. If it is different than this value, you might configure it using **DefaultSEPMRestApiPort** paramater under **Configuration** > **Parameters**.

Integration Guide for Symantec Managed Security Services (MSS)

Integration Overview

Symantec Managed Security Services (MSS) provides its customers security monitoring and real-time security analytics services including strategic insights needed to prioritize and respond to incidents and build strategies to protect the assets, reputations and viability of their organizations.

Integration Capabilities

SOAR has the following integration capabilities with Symantec MSS:

- Ingest Incident Records as Alert
- Update MSS incident record
- Close MSS incident

Use Case #1: Investigating and Mitigating Cyber-attacks

Integrated with Symantec MSS, ATAR periodically collects new incidents and update the statuses of the open incidents as they change in Smyantec MSS system. When an incident record is created on Symantec MSS, ATAR automatically collects Incident Details such as Analyst Comment, Signatures that are triggering this alert, Comments that are added to the incident and possible Attachments inside this alert anc creates a new incident on its own Incident Management Service Desk.

Configuration

Prerequisites

- SOAR connects to Symantec MSS API via HTTPS. So access permission to https://api. managedsecurity.com is required.

- A user account and a certificate-password pair are required for ATAR to connect to Symantec MSS API.

Configuring Symantec MSS

The Symantec MSS service uses client-side certificates for authentication.

1. Click **Profile** > **Certificates** > **Create a certificate.**
2. Select the **type of service** for the certificate.
3. Set the expiration date for the certificate. The available values are 6 months, 1 year, and 2 years.
4. [Optional] Specify the name for the certificate.
5. Click **Register**.

> The certificates are enabled by default upon creation, but must be downloaded and installed before they can be used.

Configuring SOAR

To use the client-side certificate created on Symantec MSS, you must convert it with **openssl** command line tool as following:

```
openssl pkcs12 -in <certificate_created_in_MSS_Portal>.p12 -clcerts -nodes -out <output_file>
```

Configuring Credentials

1. Click **Configuration** > **Credentials** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor**:
3. **Internal Credential**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Symantec MSS Credentials)).
Username	Empty
Password	Empty.
Private Key	Paste the content of the <output_file>.pem file into the Private Key area.

> The external credential stores can not be used with this integration type.

Configuring Symantec MSS as Alert Source

1. To add a new incident severity configuration, click **Configuration** > **Incidents** > **Severities** .

   Symantec MSS integration requires the following incident severity definitions:

   - Informational

   - Warning

   - Critical

   - Emergency

2. To add a new incident statues configuration, click **Configuration** > **Incidents** > **Statuses**.

   Symantec MSS integration requires the following incident status definitions:

   - New

   - In Progress as Open statuses

   - False Positive

   - Resolved

   - Deferred

   - No Action as closed statuses.

3. Click **Configuration** > **Alert Source** > **Create Alert Source Configuration.**

4. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Symantec MSS Alert Source on SOAR
Type	Symantec MSS
Address	Address of Symantec MSS service (in the following format: https://api.monitoredsecurity.com).
Alert Severities	Mapping of alert severity values to SOAR incident severities.

Parameter	Value
Configuration	Specify the following configuration parameters: # Enables incident sync # Default: false #incident.autoSync=true # Request timeout in minutes # If not provided, ATAR will use 10 by default #request.timeout=10 # Enable auto closing ATAR incidents when the related Symantec MSS incident is closed, # Default: false #incident.autoClose=true # Enable auto reopening ATAR incidents when the related Symantec MSS incident is reopened, # Default: false #incident.autoReopen=true # Scope fields to be extracted from base events and/or correlated events (field1:CATEGORY:ROLE, # CATEGORY is any of: EMAIL_ADDRESS, HASH, HOST, MAC_ADDRESS, NETWORK_ADDRESS, # COMPUTER_NAME, UNKNOWN, URL, USERNAME, PROCESS # ROLE is any of: OFFENDER, IMPACT, RELATED # # Note: The fields in the baseevent.scope example below are always extracted by default. # Note: Extraction with same field name overrides the default one. # Note: Extraction with different field name does not override the default behaviour and extracted # Note: Field names must start with / character # # Example: baseevent.scope=/sourceIPString:NETWORK_ADDRESS:OFFENDER # baseevent.scope= # # Example: correlated.scope=/sourcev6:NETWORK_ADDRESS:OFFENDER # correlated.scope= # How far (in days) into the past ATAR will look for remote incidents at the initial sync task # If not provided, ATAR will use 14 days by default #days.to.look.back.at.initial.sync=14
Credential	Name of the credential set you have created (For example, Symantec MSS Credentials).
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or is not recognized by browsers
Viisible Alert Fields	Select alarm fields that has to be displayed on Incident Management Service Desk.
Notify	Select user(s) from the list to notify when ATAR performs an action on this integration.

5. Click **Test** to test the integration.

6. Click **Save** to complete the integration.

Configuring Symantec MSS as an Integration

1. Click **Configuration** > **Integrations** > **Create Integration**.

2. Specify the following parameter values in the **Configuration** form:

Parameter	Value
Name	Display name of Symantec MSS integration on SOAR
Type	Symantec MSS
Address	Address of Symantec MSS service (in the following format: https://api.monitoredsecurity.com).
Configuration	Specify the following configuration parameters: `#proxy.id=5422`
Credential	Name of the credential set you have created (For example, ArcSight ESM Credentials).
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or is not recognized by browsers
Require Approval From	Select user(s) from list to ask her/his approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when ATAR performs an action on this integration.

3. Click **Test** to test the integration.

4. Click **Save** to complete the integration.

Additional Notes

The following configuration parameters can be used for fine tuning the integration.

Consult SOAR field engineering team before editing them:

Parameter Name Description Default Value

```
SymantecMssListenerMaxRetrySeconds Symantec MSS listener queue max message
retry in seconds 1800
SymantecMssListenerQueueConcurrency Upper limit of Symantec MSS Listener
consumer thread count 3
SymantecMssSyncLookBehindMinutes Minutes to look behind to incident in
Symantec MSS SyncTask 20
```

```
SymantecMssSyncPeriod Period in seconds to sync Symantec MSS incidents 60
Below Automation Bit sample could be used to automatically close incidents via
Trigger.
atar.require("underscore");
var remoteStatusList = [
'False Positive',
'Resolved',
'Deferred',
'No Action'
];
var remoteStatus = 'Resolved';
var statusName = atar.getTicket().getTicketStatus().getName();
if (_.contains(remoteStatusList, statusName)) {
remoteStatus = statusName;
}
var params = {'INCIDENT_CLOSING_STATUS': remoteStatus};
atar.action(ActionPluginCapability.CLOSE_INCIDENT, atar.getAlert(),
atar.device("Symantec MSS Integration"), params);
```

Integration Guide for Symantec Messaging Gateway

Integration Overview

Symantec Messaging Gateway (Brightmail) is an email gateway which is used to filter incoming and outgoing emails.

Integration Capabilities

SOAR has the following integration capabilities with Symantec Messaging Gateway:

- Block Sender
- Block in Dictionary

Use Case: Blocking phishing attacks

SOAR can follow the email inboxes for user's phishing reports and automatically creates an incident record on its service desk. To stop the phishing campaigns, SOAR can extract the sender address, IP, e-mail subject and block them on Symantec Messaging Gateway.

This can be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Currently SOAR supports Symantec Messaging Gateway 10.6.5-1 version.
- SOAR connects to Symantec Messaging Gateway via HTTPS. Access to 443/tcp port is required.
- A user account for SOAR to connect Symantec Messaging Gateway.

Configuring Symantec Messaging Gateway

1. Click **Administration** > **Users** and select **Create a new administration policy** to create an administrator account. Select **Manage Policies right**.

   Disable all other rights since they are unnecessary.

2. Click **Content** > **Dictionaries** to create a dictionary.

3. To block hosts and IP addresses, SOAR uses **Local Bad Sender IPs** and**Local Bad Sender Domains**.

Configuring SOAR

1. Navigate to **Configuration** > **Credentials** and click **Create Credential.**
2. Fill the **Credential Editor** form with following parameter values:

   a. **Internal Credential:**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Symantec Messaging Gateway Credential)
Username	Username you have created of SOAR on Symantec Messaging Gateway
Password	Password of the user you have created of SOAR on Symantec Messaging Gateway.
Private Key	Empty

   b. Credential Store

Parameter	Value
Type	External Credential
Name	Name of the credential with full path of the safe on store

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Fill the configuration form with the following parameter values:

Parameter	Value
Name	Display name of Symantec Messaging Gateway integration on SOAR.
Type	Symantec Messaging Gateway.
Address	Address of the integration (the format must be192.168.2.212).)
Configuration	You need to specify the following configuration parameters. You can define multiple dictionaries by seperating "|", for example, dictionary.name=SOAR Dictionary 1	SOAR Dictionary 2
Credential	Name of the credential set you've just created on step 2 (for example, Symantec Messaging Credential.

Parameter	Value
Trust Invalid SSL Certificates	Select this if Symantec Messaging Gateway's certificate is self-signed or not recognized by browserss.
Require Approval From	Select user(s) from list to ask her/his approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Test** to test the integration.

6. Click **Save** to complete integration.

Integration Guide for Tenable Nessus

Integration Overview

Tenable Nessus is a vulnerability scanner used to detect vulnerabilities on the network. SOAR uses Tenable Nessus to gather vulnerability information to enrich incidents' context.

Integration Capabilities

- Action
- Get Scan List
- Get All Vulnerabilities on a Scan

Configuration

ConfiguringTenable Nessus

- SOAR connects to Tenable Nessus' API via HTTPS. Therefore SOAR must be able to connect this service.
- A user credential is required.

Configuration on SOAR

Configuring SOAR

1. Navigate to **Configuration** > **Integrations**.
2. In the **Integrations Editor**, specify the following parameter values:

Parameter	Value
Name	Display name of Tenable Nessus integration on SOAR
Type	Tenable Nessus.
Address	Address of the integration (in the following format: http[s]://1.1.1.1:1234 or http[s]://abc.example.com:1234
Credential	Credential defined for the integration under the **Credentials** menu

Parameter	Value
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

3. Click **Test** to test the integration.

4. Click **Save** to complete the integration.

Integration Guide for Tenable Security Center

Integration Overview

Tenable Security Center (Tenable SC) is a vulnerability management solution that provides visibility into network by identifying all vulnerabilities, misconfigurations and malware attack on assets and gives ability to manage and measure your cyber risk.

SOAR has the following integration capabilities with Tenable Security Center:

- Get Assets
- Get Vulnerabilities (System-wide)
- Get Vulnerabilities on IP

.Use Case: Getting vulnerability details of assets

SOAR can integrate with Tenable Security Center to gather additional information about an asset during incident investigation. Knowing existing vulnerabilities on a system can help SOC analysts to understand possible root cause of an incident more precisely.

Configuration

Prerequisites

- SOAR connects to Tenable Security Center's API using HTTPS. Typically an access permission to 443/tcp port is required.
- A user account for SOAR to connect to Tenable Security Center.

Configuring Tenable Security Center

1. Login to Tenable Security Center with Security Manager User.

   > Note: This user account is different from admin account.

2. Navigate to **Users**> **Groups** and add a group to define the objects that SOAR can access. You must at select atleast one item from **Viewable Hosts and Repositories lists**.

   There is no need to share any object under **Share to Group** tab.

3. To add user for SOAR access, navigate to **Users** > **Users**. Select **No Role** and **SOAR Access Group** in **Membership**.

Configuring SOAR

1. Navigate to **Configuration** > **Credentials** and click **Create Credential.**
2. Fill the **Credential Editor** form with following parameter values:
   a. **Internal Credential:**

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Tenable SC Credential
Username	User you have created of SOAR on Tenable Security Center.
Password	Password of the user you have created of SOAR on Tenable Security Center.
Private Key	Empty

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Fill the configuration form with the following parameter values:

Parameter	Value
Name	Display name of Tenable Security Center integration on SOAR.
Type	Tenable Security Center.
Address	Address of the integration (the format must be https://1.1.1.1:1234 or https://abc.example.com:1234)
Credential	Name of the credential set you've just created on step 2 (for example, Tenable SC Credential.
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or not recognized by browserss.
Require Approval From	Select user(s) from list to ask her/his approval before executing actions on this integration.
Notify	elect user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Test** to test the integration.
6. Click **Save** to complete integration.

Integration Guide for Trend Micro Apex Central

Integration Overview

Trend Micro Apex Central is a web-based console that provides centralized management for Trend Micro products and services at the gateway, mail server, file server and corporate desktop levels.
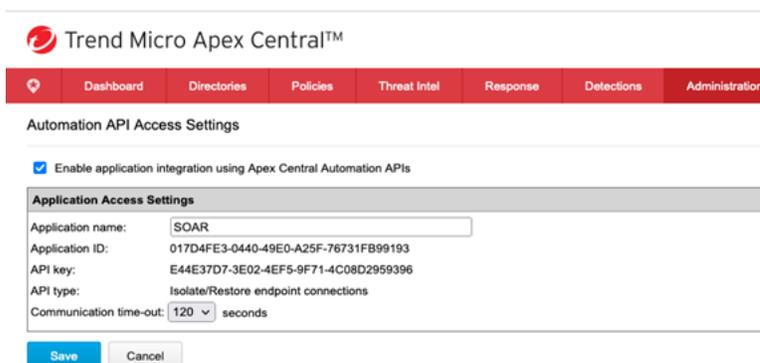
Integration Capabilities

- Quarantine

Prerequisites

- Access to the HTTPS is needed as ArcSight SOAR connects to Trend Micro Apex Central API through this service.

Configuring Trend Micro Apex Central

1. Login to Trend Micro Apex Central and navigate to **Administration** tab.
2. Click **Settings** < **Automation API Access Settings** and add a new application as follows:

3. Note down the **Application ID** and **API Key** (for your reference later) after saving the

application as follows:
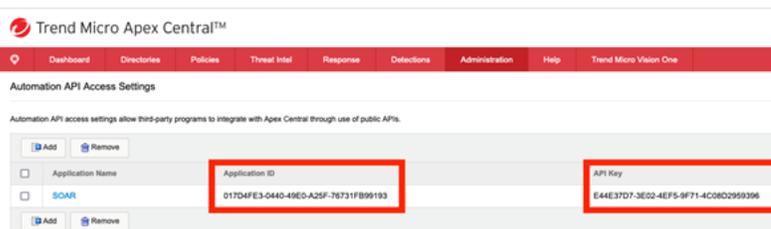
Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credentials**.

2. Specify the following parameter values in the **Credential Editor**:

   Internal Credential:

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Trend Micro Apex Central Credentials)
Username	Empty
Password	Application ID that you've noted from created application.
Private Key	API Key that you have noted before from the created application.

3. Click **Configuration** >**Integrations**> **Create Integration.**

   Specify the following parameter values in the **Configuration** form.:

Parameter	Value
Name	Display name of Trend Micro Apex Central integration on SOAR
Type	Trend Micro Apex Central
Address	Address of the integration (the format must be https://czbxlz.manage.trendmicro.com)
Credential	Name of the credential set that you created on step 2. (For example, Trend Micro Apex Central Credentials).

Parameter	Value
Trust Invalid SSL Certificates	Select this if Trend Micro Apex Centrals certificate is self signed or it is not recognized by browsers.
Require Approval Form	Select user(s) from list who can provide approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

4. Click **Save** to save the integration definition.

5. Navigate to **Configuration>Customization Library** and edit **Trend Micro Apex Central Advanced Action Script Default Script Template**.

6. Select the integration that you have added to **Integrations** menu.

7. Click **Save** to complete the integration.

8. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid

Capablities

Quarantine

Action capability for quarantine a Hostname, IP address and MAC addresss.

- Rollback: Yes
- Duplicate Control: No

Input Parameter	Description	Type	Scope Restricted Yes/No	Required Yes/No
Rollback Mode	Time to rollback this action. Default is no-rollback.	N/A	N/A	No
MAC Address / Network Address / Hostname	MAC Address/Network Address/Hostname to quarantine	MAC Address Network Address Hostname	Yes	Yes

Output:

Case Scope: N/A

Human Readable Output: N/A

Integration Guide for Trend Micro Vision One

Integration Overview

Trend Micro Vision One is a purpose-built threat defense platform that provides added value and new benefits beyond XDR solutions, allowing you to see more and respond faster.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Trend Micro Vision One:

- Get Observed Attack Techniques
- Query Operating Systems
- Get Exception List
- Get Suspicious Object List
- Add Objects to Suspicious Object List
- Add Objects to Exception List
- Delete Objects from Suspicious Object List
- Delete Objects from Except List
- Prerequisites

You must have access to HTTPS as the ArcSight SOAR connects to Trend Micro Vision One to API through this service.

Configuration

Configuring Trend Micro Vision One

1. Login to the **Vision Platform** and create a user with the **Master Administrator** role and **Trend Micro Vision One™ console** and **APIs** access level.
2. Get access token of the created user that is used as a credential on ArcSight SOAR.

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**.

2. Specify the following parameter values in the **Credential Editor** form.

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, Trend Micro Vision One Credential).			Bearer<space><access-token>

3. Click **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration** form.

Parameter	Value
Name	Display name of the integration.
Type	Trend Micro Vision One.
Address	URL of API (for example, API trend micro).
Configuration	Specify the following configuration parameters:

cache.reusing.duration	Configure how far (in minutes) into the past this enrichment will look. For example: cache.reusing.duration=20 .
proxy.id	ID of the proxy integration when you access Trend Micro Vision One through a web proxy device. For example, proxy.id = 12345 .

Credential	Credential that has been defined for this integration under the **Credentials** menu.
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Save** to save the integration definition.

6. Navigate to **Configuration**>**Customization Library** and edit **Trend Micro Vision One Advanced Action Script Default Template**.

7. Select the integration that you have added to **Integrations** menu.

8. Click **Save** to complete the integration.

9. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Get Observed Attack Techniques**

   Enrichment capability for getting observed attack techniques.

   The following table presents the **Get Observed Attack Techniques** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	No	Yes
Risk Levels	Single or comma seperated risk values (high, critical, low, info, undefined, medium).	Text	No	No
Tactic IDs	Single or comma seperated tactid id values .	Text	No	No
Technique IDs	Single or comma seperated technique id values.	Text	No	No
Name Filter	Detection Filter name .	Text	No	No
Endpoint Name	Name of the endpoint.	Computer Name, Hosy, Keyword, Unknown	Yes	No
Time Range	Time range for attack times.	Time Range	No	Yes

   Output:

   Case Scope:

Action	Type	Category/ Value
Add	Scope Item	Keyword (Related)

   Human Readable Output:

2. **Query Operating Systems**

   Enrichment capability for operating system information for all agents active in the last seven days.

   The following table presents the **Query Operating Systems** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	No	Yes

Output:

Case Scope:

Action	Type	Category/ Value
Add	Scope Item	Keyword (Related)

Human Readable Output:

3. **Get Exception List**

   Enrichment capability for information about domains, file SHA-1 values, IP addresses, or URLs that are in the Exception List.

   The following table presents the **Get Exception List** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	No	Yes
Type	Single or comma seperated types ("domain", "ip", "sha1", "url").	Text	No	No

   Output:

   Case Scope:

Action	Type	Category/ Value
Add	Scope Item	Keyword (Related)

   Human Readable Output:

4. **Get Suspicious Object Lists**

   Enrichment capability for information about domains, file SHA-1 values, IP addresses, or URLs that are in the Suspicious Object List

   The following table presents the **Get Suspicious Object** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	No	Yes
Type	Single or comma seperated types ("domain", "ip", "sha1", "url").	Text	No	No
Content Filter	Filters the list to suspicious objects that exactly match the specified string.	Text	No	No

Output:

Case Scope:

Action	Type	Category/ Value
Add	Scope Item	Keyword (Related)

Human Readable Output:

5. **Add Objects to Suspicious Object List**

   Action capability for Adding domains, file SHA-1 values, IP addresses, or URLs to the Suspicious Object List.

   • Rollback: Yes

   • Duplicate Check: No

   The following table presents the **Add Objects to Suspicious Object List** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Rollback Mode	Time to rollback this action. Default is no-rollback.	Relative Time	No	No
Value	The value of the suspicious object which will be added.	Host, Network Address, Hash, URL	Yes	Yes
Description	Record description info.	Text	No	No

Input Parameter	Description	Type	Scope Restricted (Yes/No)	Required (Yes/No)
Scan Action	Suspicious object record scan action, when not set, use system default settings. Risk Level. Type's scan action.	Enum	No	No
Risk Level	Suspicious object risk level when not set, use default value - high.	Enum	No	No
Expired Day	Suspicious object record expired day, when not set, use system default settings. Expired Day.	Text	No	No

Output:

Case Scope:

Action	Type	Category/ Value
Add	Scope Item	Keyword (Related)

Human Readable Output: N/A

6. **Add Objects to Exception List**

Action capability for Adding domains, file SHA-1 values, IP addresses, or URLs to the Exception List and prevents these objects from being added to the Suspicious Object List.

- Rollback: Yes

- Duplicate Check: No

The following table presents the **Add Objects to Exception List** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Rollback Mode	Time to rollback this action. Default is no-rollback.	Integration	No	Yes
Value	Suspicious object record value,it support full match or partial match, DOMAIN partial match: (with a wildcard before 1st, example, example.com) IP partial match: (ip range example, 200.102.35.1-200.102.35.254,cidr example: 200.102.35.1/24) URL Partial match: (support wildcard 'http://.'', 'https://.'' at beginning, or ''' at the end, or both two wildcards, example, https://.example.com/path1/) SHA1 (only full match).	Text	No	No
Description	Exception description info.	Text	No	No

Output:

Case Scope:

Action	Type	Category/ Value
Add	Scope Item	Keyword (Related)

Human Readable Output: N/A

7. **Delete Objects from Exception List**
   Action capability for Deleting domains, file SHA-1 values, IP addresses, or URLs from the Exception List.

   • Rollback: Yes

   • Duplicate Check: No
     The following table presents the **Delete Objects from Exception List** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Rollback Mode	Time to rollback this action. Default is no-rollback.	Relative Time	No	No
Value	Suspicious object record exception value.	Host, Network Address, Hash, URL	Yes	Yes

Output:

Case Scope:

Action	Type	Category/ Value
Add	Scope Item	Keyword (Related)

Human Readable Output: N/A

8. **Delete Objects from Suspicious Object List**
   Action capability for Deleting domains, file SHA-1 values, IP addresses, or URLs from the Suspicious Object List:

   • Rollback: Yes

   • Duplicate Check: No

The following table presents the **Delete Objects from Suspicious Object List** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Rollback Mode	Time to rollback this action. Default is no-rollback.	Relative Time	No	No
Value	Suspicious object record exception value.	Host, Network Address, Hash, URL	Yes	Yes

Output:

Case Scope:

Action	Type	Category/ Value
Add	Scope Item	Keyword (Related)

Human Readable Output: N/A

Integration Guide for Turkcell Threat Intelligence

Integration Overview

Turkcell Threat Intelligence is a service which lets users to query reputation of Indicators of Compromise such as data leakage, brand protection, and vulnerability modules.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Turkcell Threat Intelligence:

- Domain Query
- Email Query
- Hash Query
- IP Query

Use Case: Investigating Phishing Campaigns

SOAR integrates with Turkcell Threat Intelligence or Bozok to investigate and mitigate phishing campaigns. SOAR extracts the indicators such as sender address, IP address, and URLs from a phishing report email of the user and creates a new incident on the Incident Management Service Desk. SOAR then checks with Turkcell Threat Intelligence or Bozok if this is a known attack and previously analyzed. This investigation can either be performed automatically within a playbook or manually by an analyst.

Configuration

Prerequisites

- Access to https://bozok.turkcell.com.tr (443/tcp port) as SOAR connects to Turkcell Threat Intelligence/Bozok API through HTTPS
- An API key for SOAR to connect to Turkcell Threat Intelligence/Bozok service

Configuration on Turkcell Threat Intelligence or Bozok

- No specific configuration is needed on Turkcell Threat Intelligence or Bozok.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor form**:

   a. Internal Credential:

Parameter	Value
Type	Internal Credential
Name	Display name of credential set (For example, Turkcell Threat Intelligence Credentials)
Username	Empty
Password	Empty
Private Key	API key obtained from the service provider

   b. Credential Store:

Parameter	Value
Type	External credential
Name	Name of the credential with full path of the safe on store

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Specify the following parameter values in the **Configuration form**:

Parameter	Value
Name	Display name of Turkcell Threat Intelligence integration on SOAR
Type	Turkcell Threat Intelligence
Address	Address of Turkcell Threat Intelligence service(in the following format: https://bozok.turkcell.com.tr)
Credential	Name of the credential set created (For example, Turkcell Threat Intelligence Credentials)
Trust Invalid SSL Certificates	Unselect

Parameter	Value

Configuration	Specify the following configuration parameters:
	
	``
	# Integration ID of the proxy integration to use
	when connecting to current integration.
	# If not provided, SOAR will try to use a direct connection.
	proxy.id=5434
	# configure how far (in minutes) into the past
	this enrichment will look.
	cache.reusing.duration=60
	``

Require Approval From	Not applicable as SOAR executes enrichment on Turkcell Threat Intelligence
Notify	Not applicable as SOAR executes enrichment on Turkcell Threat Intelligence

5. Click **Test** to test the integration.

6. Click **Save** to save the integration.

Integration Guide for Udger

Integration Overview

Udger is a query detection repository service that works for both cloud-based and local executions. Udger also provides Data Center name of given IP and many more.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with Udger:

- Parse

Prerequisites

- You must have access to HTTPS as the ArcSight SOAR connects to Udger API through this service.
- API Token is needed to access Udger REST API.

Configuration

Configuring Udger

1. Login to udger and navigate to **Products** > **Cloud Parser**.
   - Click **Subscribe Now**.

   - Select **Subscription Package** > **Activate**.

     > Note: You can activate free package for the trial usage

   - The access key is displayed in **My Account** > **General**

     > Note: Copy the access key as this is required during creating credential.

Configuring SOAR

1. Click **Configuration** > **Integration** > **Create Integration**.
2. In **Configuration Editor**, select **Udger** in **List of Type**.

3. Navigate to **Credential** and click **Create** to create new credential. Specify following values in the **Credential Editor**:

Type	Username	Password	Private Key	Check
Internal Credential			**Access Key** that is copied from **Udger** web site (navigate to **My Account** > **General** tab on Udger UI).	**Clear Text Access** checkbox.

4. Click **Save** to save the integration definition.

5. Navigate to **Configuration**>**Customization Library** and edit **Udger Advanced Action Script Default Template**.

6. Select the integration that you have added to **Integrations** menu.

7. Click **Save** to complete the integration.

8. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Parser**
   Parsing given IP and return JSON detail including Datacenter Name

   The following table presents the **Parser** capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
IP	A valid IP Address to retrieve data.	Network Address Host	Yes	Yes
User Agent	User Agent information to query on IP.	Unknown Keyword	Yes	No

   Output:

   Case Scope:

   Scope Item Property **Cloud Name** is added to the related scope item. When you click the related scope item to view its properties, a cloud name result is displayed.

   The following table presents the case scope details:

Action	Type	Category/ Value
Set	Scope Item Property	Cloud Name

   Human Readable Output:

Integration Guide for USOM (TR-CERT) Intelligence Feed

Integration Overview

USOM (TR-CERT- Computer Emergency Response Team of the Republic of Turkey) Intelligence Feed is an actively maintained local feed about various malicious categories prepared by TR-CERT team.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with USOM (TR-CERT):

- Ingest Feed as Alert

Use Case: Blocking malicious URLs and IPs before they do any harm

SOAR integrates with USTA(**need the acronym in this context**) (TR-CERT) intelligence feed to block malicious entities on your perimeter protection before they do any harm.

Configuration

Prerequisites

- Access to https://www.usom.gov.tr/url-list.xml as SOAR connects to USTA (TR-CERT) intelligence feed through HTTPS.

Configuring USOM (TR-CERT) Intelligence Feed

- No specific configuration is needed on USOM (TR-CERT) Intelligence Feed.

Configuring SOAR

Configuring USOM (TR-CERT) Intelligence Feed as Alert Source

1. Click **Configuration** > **Alert Source**> **Create Alert Source Configuration**.
2. Specify the following parameter values in the **Configuraiton form**:

Parameter	Value
Name	Display name of USOM (TR-CERT) Intelligence Feed Alert Source on SOAR
Type	USOM (TR-CERT)
Address	Address of USOM (TR-CERT) Intelligence Feed (in the following format: https://www.usom.gov.tr/url-list.xml)

Configuration	Specify the following configuration parameters:
	
	``
	# Ignore events older than specified date.
	If empty, date based filtering is disabled.
	# Example: filterOlderThanDate=2017-01-01
	filterOlderThanDate=2019-08-01
	# Integration ID of the proxy integration to use
	when connecting to current source.
	# If not provided, SOAR will try to use a direct
	connection.
	#proxy.id=2443
	``

Trust Invalid SSL Certificates	Unselect
Visible Alert Fields	Define which alarm fields are displayed Incident Management Service Desk
Notify	Not applicable

3. Click **Test** to test the integration.
4. Click **Save** to save the integration.

Additional Notes

The intelligence feed is specialized for Turkey and is accessible only from IP ranges of Turkey.

Integration Guide for Urlscan

Integration Overview

The **URLscan** API allows you to submit URLs to scan, retrieve scan results, download Document Object Model (DOM) snapshots and page screenshots and search existing scans for different types of indicators.

Integration Capabilities

ArcSight SOAR has the following integration capabilities with urlscan:

- Search Domain
- Search Hash
- Search IP
- Search URL
- Submit URL

Configuration

Prerequisites

- You must have access to HTTPS as the ArcSight SOAR connects to urlscan io API through this service.
- URLScan requires an API key for access.

Configuring SOAR

1. Click **Configuration** > **Credential** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor** form:

Type	Name	Username	Password	Private Key
Internal credential	Display name of credential set (for example, URL Scan API Credential).	Empty	Empty	Access Token

3. Click **Configuration** > **Integrations** > **Create Integration**.
4. Specify the following parameter values in the **Configuration** form.

Parameter	Value
Name	Display name of the integration.
Type	Urlscan.io
Address	Address of the integration (the format must be https://urlscan.io).
Configuration	Specify the following configuration parameters: **proxy.id** **ID of the proxy integration if you access Urlscan.io through a web proxy device. For example: proxy.id = 12345 .**
Credential	Credential that has been defined for this integration in the **Credentials** menu.
Trust Invalid SSL Certificates	Select this if web server's certificate is self-signed or is not recognized by browsers.
Require Approval From	Select user(s) from list to ask the approval before executing actions on this integration.
Notify	Select user(s) from the list to notify when SOAR performs an action on this integration.

5. Click **Save** to save the integration definition.

6. Navigate to **Configuration**>**Customization Library** and edit **Urlscan Advanced Action Script Default Template**.

7. Select the integration that you have added in the **Integrations** menu.

8. Click **Save** to complete the integration.

9. Click **Test**, an **Integration Successful** message is displayed if the credential and address are valid.

Capabilities

1. **Search Domain**

   Enrichment capability for retrieving domain information for a relative time range.

   The following table presents the **Search Domain** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Bucket Name	Name of the third party integration.	Integration	N/A	Yes
Domain	Domain to be queried from Urlscan.	Host	Yes	Yes

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Relative Time Range	Specific relative time period that will be checked.	Time unit Hour (s) Day(s) Week(s) Month(s)	N/A	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

2. **Search Hash**

   Enrichment capability for retrieving hash information for a relative time range.

   The following table presents the **Search Hash** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
SHA-256	SHA-256 hash value to be queried from Urlscan.	Host	Yes	Yes
Relative Time Range	Specific relative time period that will be checked.	**Time unit** Hour (s) Day(s) Week(s) Month(s)	N/A	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

3. **Search IP**

Enrichment capability for retrieving IP information for a relative time range.

The following table presents the **Search IP** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
IP	Network address to be queried from Urlscan.	Network Address	Yes	Yes
Relative Time Range	Specific relative time period that will be checked.	**Time unit** Hour (s) Day(s) Week(s) Month(s)	N/A	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

4. **Search URL**

Enrichment capability for retrieving URL information for a relative time range..

The following table presents the **Search URL** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
URL	URL to be queried from Urlscan.	URL	Yes	Yes

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Relative Time Range	Specific relative time period that will be checked.	**Time unit** Hour(s) Day(s) Week(s) Month(s)	N/A	Yes

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

5. **Submit URL**

Enrichment capability for submitting a URL for investigation.

The following table presents the **Submit URL** action capability details:

Input Parameter	Description	Type	Scope Restricted (Yes/ No)	Required (Yes/ No)
Integration	Name of the third party integration.	Integration	N/A	Yes
URL	URL to be queried from Urlscan.	URL	Yes	Yes
Tag	User-defined tags to annotate this scan, for example, phishing or malicious. Limited to 10 tags.	String	N/A	No
Visibility	Submitting visibility option which could either be Public, Private or Unlisted	String Public Unlisted Private	N/A	Yes
Do not Use Cache	If this option is checked, SOAR does not use cached results.	Boolean	N/A	No

Output:

Case Scope:

Enrichment	Type	Category Value
None	N/A	N/A

Human Readable Output:

Field	Value
categories	
score	0
page_server	nginx
page_url	https://www.microfocus.com/en-us/home
page_asnname	AKAMAI-ASN1, NL
page_ptr	a104-126-37-176.deploy.static.akamaitechnologies.com
page_ip	104.126.37.176
page_domain	www.microfocus.com
page_asn	AS20940
page_country	DE
page_city	Frankfurt am Main

Integration Guide for VirusTotal

Integration Overview

VirusTotal inspects suspicious files and URLs to detect types of malware with over seventy antivirus scanners and URLs or domain blacklisting services, in addition to a myriad of tools to extract signals from the studied content.

Integration Capabilities

SOAR has the following integration capability with VirusTotal:

- Domain Query
- Domain/Downloaded Files Query
- Domain/Subdomains Query
- Domain/URLs Query
- File Query
- Hash Query
- IP Query
- IP/Downloaded Files Query
- IP/Passive DNS Query
- IP/URLs Query
- URL Query

Use Case: Blocking access to malicious URL

During the investigation of an attack, SOAR checks for suspicious IP addresses, URLs, files, and hash values to VirusTotal if these indicators are known and previously analyzed. According to returned confidence score, SOAR decides on the next course of action. This investigation can either be performed automatically within a playbook or manually by an analyst.
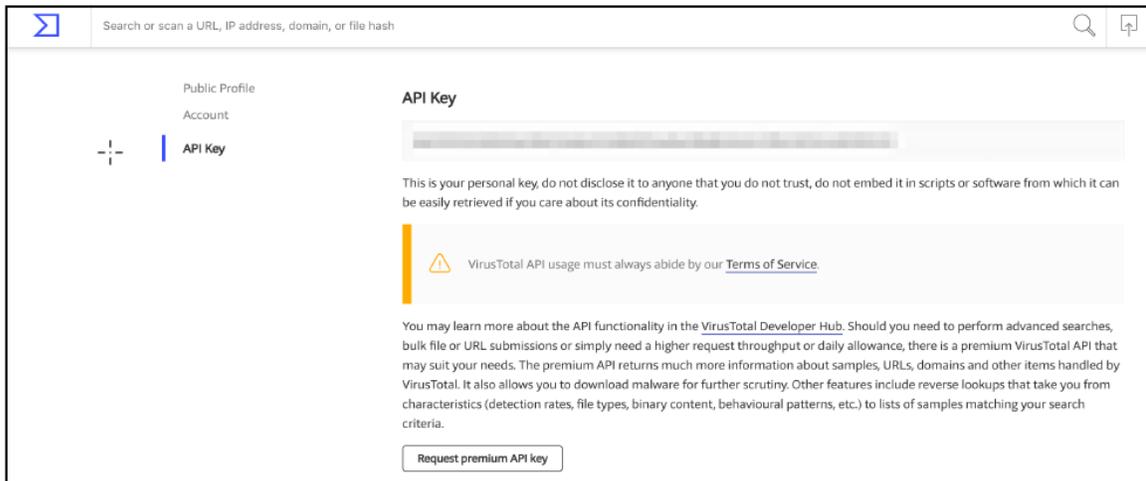
Configuration

Prerequisites

- VirusTotal API version 3
- Access to tcp port 443 as SOAR connects to VirusTotal API http://www.virustotal.com

- An API key for SOAR to connect to VirusTotal

Configuring VirusTotal

- No specific configuration is needed on VirusTotal.
- Login to https://www.virustotal.com with your username and make a note of the API key under **Settings**> **API Key**.

Configuring SOAR

1. Click **Configuration** > **Credentials** > **Create Credential**.
2. Specify the following parameter values in the **Credential Editor**:

   a. **Internal Credential**

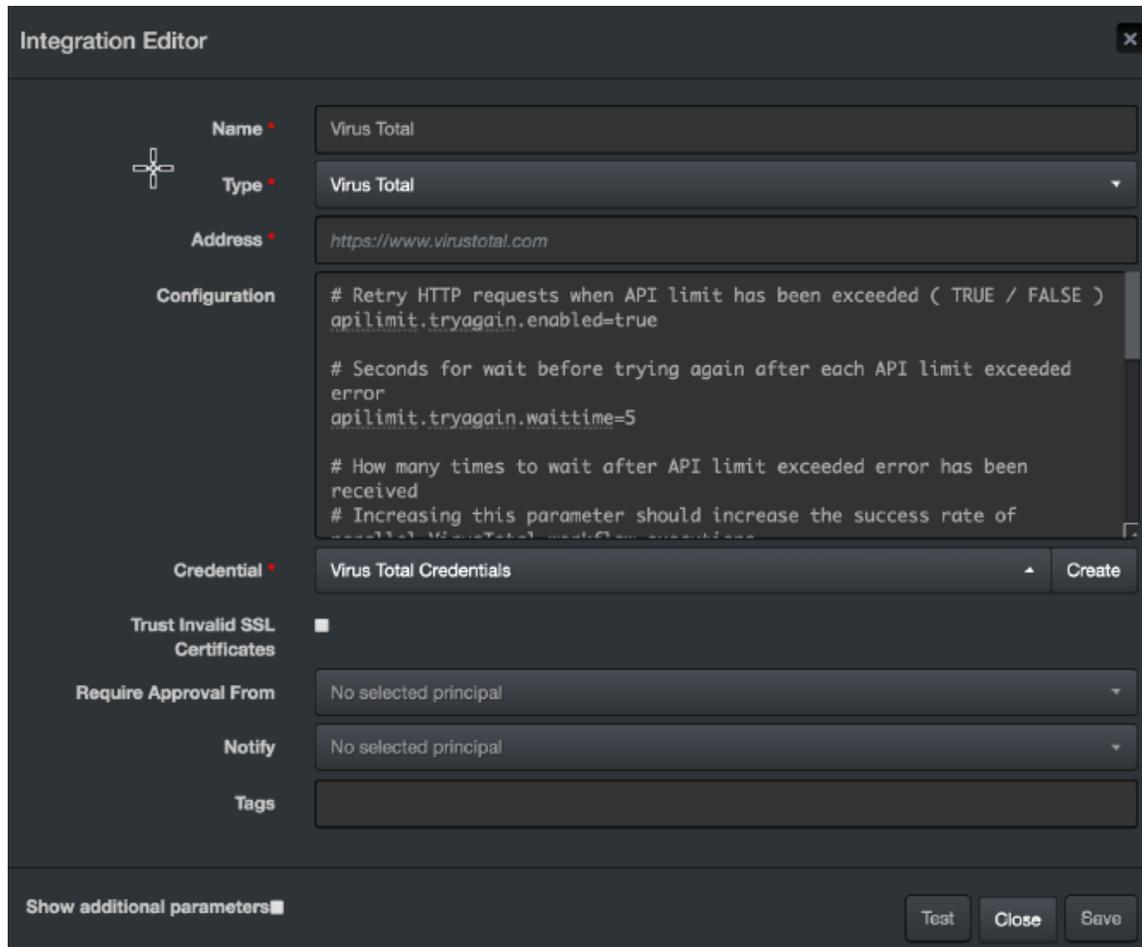 | Parameter | Value |
   |---|---|
   | Type | Internal Credential |
   | Name | Display name of credential set (For example, VirusTotal Credentials) |
   | Username | Empty |
   | Password | Empty |
   | Private Key | API Key you have on VirusTotal |

   b. **Credential Store:**

Parameter	Value
Type	External Credential
Name	Name of the credential with pull path of the safe on store

3. Navigate to **Configuration** > **Integrations** > **Create Integration**.

4. Specify the following parameter values in the **Configuration form**:

Parameter	Value
Name	Display name of VirusTotal integration on SOAR
Type	VirusTotal
Address	Address of the integration (in the following format https://www.virustotal.com)
Configuration	Specify the following configuration parameters:

```
# Retry HTTP requests when API limit has
been exceeded ( TRUE / FALSE )
apilimit.tryagain.enabled=true
# Seconds for wait before trying again
after each API limit exceeded error
apilimit.tryagain.waittime=5
apilimit.tryagain.waittime=5
# How many times to wait after API limit
exceeded error has been received
# Increasing this parameter should increase
 the success rate of parallel VirusTotal workflow
apilimit.tryagain.waitlimit=3
# Integration ID of the proxy integration
to use when connecting to current integration.
# If not provided, ATAR will try to use a
direct connection.
#proxy.id=123
# configure how far (in minutes) into
the past this enrichment will look.
#cache.reusing.duration=20
# Enrichment timeout duration after
 start time ( in seconds

)
scan.query.timeout=3600
# Expiration period of hash scans
# If not provided, ATAR will use 30 days
by default
#scan.result.expiration.period.in.days=30
# VirusTotal APIv3 parameter
# Limits page count for relation queries.
  SOAR will use 1 page by default
#scan.result.page.count.max=1
```

Parameter	Value
Trust Invalid SSL Certificates	Unselect
Require Approval From	Not applicable
Notify	Not applicable

5. Click **Test** to test the integration.

6. Click **Save** to complete the integration.

Additional Notes

- Domain and IP-related queries retrieve results in 40-item batches. For some IOCs, this may result in too many consecutive queries and long query-times.

- The file queries are limited to 32MB due to limits with VirusTotal API.

- Domain or URLs, Domain or Downloaded Files, IPor URLs, and IP or Downloaded Files only return the scope items with confidence score greater than 0.

Integration Guide for VMware ESXi

Integration Overview

SOAR uses VMware ESXi(Elastic Sky X integration) to perform some actions on the virtual machines (VMs).

Integration Capabilities

Action

- Create Snapshot of a VM
- Export VM
- Get Information of All VMs
- Power On VM
- Power Off VM
- Reset VM
- Reboot VM
- Standby VM
- Suspend VM

Configuration

Configuring VMware ESXi

- Access to HTTPs for SOAR to connect to VMware ESXi Server's SDK
- SOAR account with admin role

Configuring SOAR

1. Navigate to **Configuration** > **Integrations**.
2. In the Integrations Editor, specify the following parameter values:

Parameter	Value
Name	Display name of VMware ESXi integration on SOAR
Type	VMware ESXi
Address	Address of the integration (in the following format: http[s]://1.1.1.1:1234[/sdk] or http[s]://abc.example.com:1234[/sdk])
Credential	Credential defined for the integration under the Credentials menu
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

3. Click **Test** to test the integration.
4. Click **Save** to complete the integration.

Integration Guide for VxStream Sandbox

Integration Overview

VxStream Sandbox is an automated malware analysis system that includes the unique Hybrid Analysis technology. It is available as a standalone software package that is automatically deployed within your local infrastructure and operates without an external dependency or callback mechanism. It is possible to execute files on any Windows guest image (For example, a copy of your local workstation) and has a variety of integration and interface capabilities.

The feature set of VxStream Sandbox is extensive, with hundreds of generic indicators at its core. It detects unknown threats independent of Anti-Virus signatures. Empowered by Hybrid Analysis, the entire process memory gets analyzed using multiple timed snapshots, including the runtime sample. This feature allows the extraction of more indicators (Strings/API calls) regardless of execution. This approach enables the analysis of dormant code, evasive conditions, and extracts more valuable IOCs.

Integration Capabilities

Action

- Hash analysis

Configuration

Configuration on VxStream Sandbox

- Access to HTTPs for SOAR to connect to VxStream Sandbox

Configuring SOAR

1. Navigate to **Configuration** > **Integrations**.
2. In the **Integrations Editor** window, specify the following parameter values:

Parameter	Value
Name	Display name of VxStream Sandbox integration on SOAR
Type	VxStream Sandbox
Address	Address of the integration (in the following format: https://www.hybrid-analysis.com)

Parameter	Value
Configuration	Specify the following configuration parameters: ``` # Integration ID of the proxy integration to use when connecting to # current integration. # If not provided, ATAR will try to use a direct connection. #proxy.id=123 # configure how far (in minutes) into the past this enrichment will look. #cache.reusing.duration=20 ```
Credential	Credential defined for the integration under the Credentials menu
Trust Invalid SSL Certificates	Select this if Engine's certificate is self-signed or is not recognized by browsers
Require Approval From	Select users from the list who can provide approval before executing actions on this integration
Notify	Select users from the list to notify when SOAR performs an action on this integration

3. Click **Test** to test the integration.
4. Click **Save** to complete the integration.

Integration Guide for WinRM

Integration Overview

This appendix provides a detailed, step-by-step configuration procedure to enable SOAR to properly work with WinRM.

Configuration On Domain-Controller

- **To create a Group Policy object for your domain:**

1. Navigate to **Start** > **Control Panel**.

2. In the Control Panel, select **Administrative Tools** > **Group Policy Management**.

3. From the menu tree, click **Domains** > **[your domain's name]**.

4. Right-click and select **Create a GPO in this domain, and Link it here**.

5. Input **WinRM-SOAR**.

6. Execute the following command:

```
reg add HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System /v

LocalAccountTokenFilterPolicy /t REG_DWORD /d 1 /f
```

7. Click **OK**.

- **To edit the new Group Policy object you created:**

1. Right-click on the new WinRM-SOAR Group Policy Object and select **Edit**.

2. From the menu tree, click **Computer Configuration** > **Policies**.

3. In the **Policies**, click **Administrative Templates: Policy definitions** > **Windows Components** > **Windows Remote Management (WinRM).**

4. Navigate to **WinRM Service**.

> **Note:** After editing the Group Policy Object, three WinRM service settings are enabled.

a. **To Allow remote server management through WinRM**

i. Right-click either **Allow automatic configuration of listeners(Windows Server 2010)** or **Allow remote server management through WinRM(Windows Server 2012)**

ii. Click **Edit**.

iii. To allow remote server management through WinRM, select **Enabled**.

iv. Enter an asterisk (*) in each field.

v. Click OK.

b. **To Allow unencrypted traffic through WinRM**

i. Right-click **Allow unecrypted traffic** and click **Edit**.

ii. Select **Enabled** and click **OK**.

Now the Windows Remote Management is enabled on the Group Policy.

c. **To Enable the Service that goes with it**

i. In the **Group Policy Management Editor window**, click **Preferences** > **Control Panel Settings** > **Services**.

ii. Right-click **Services and select New** > **Service**.

iii. Select **Automatic** as the startup.

iv. Enter **WinRM** as the service name.

v. Select **Start service** as the service action.

vi. Select **This account** to log in as.

vii. Enter **NT AUTHORITY\NetworkService** as the user and use **a space character** as the password.

viii. Click **OK**.

• **To allow inbound remote administration by updating the firewall rules**:

The steps enable the following firewall rules:

○ Windows Firewall: Allow inbound remote administration exception

○ Windows Firewall: Allow ICMP exception

1. In the **Group Policy Management Editor**, click **Computer Configuration** > **Policies**.

2. Click **Administrative Templates: Policy definitions** > **Network** > **Network Connections** > **Windows Firewall** > **Domain Profile**.

3. Right-click **Windows Firewall: Allow inbound remote administration exception** and click **Edit**.

4. Select **Enabled**.

5. Enter an asterisk (*) into each field and click **Ok**.

6. Right-click **Windows Firewall: Allow ICMP exception** and click **Edit**.

7. Select **Enabled**.

8. Select **Allow inbound echo request** and click **Ok**.

- **To create a new inbound firewall rule and update the network list manager for unidentified networks:**

1. Click **Computer Configuration** > **Windows Settings** > **Security Settings** > **Windows Firewall with Advanced Security** > **Inbound Rules.**

2. Right-click **Inbound Rules** and click **New Rule**.

3. Select **Predefined**.

4. Select **Windows Remote Management** from the list of services.

5. Click **Next**.

6. Unselect the entry profile **Public** and click **Next.**

7. Click **Finish**.

8. Right-click the new rule and click **Properties**.

9. Click the **Advanced** tab and unselect all and select **Private**.

10. Click the **Scope tab.**

11. Check these IP addresses on Remote IP Address and specify IP address of the SOAR machine and click **OK**.

12. From the menu tree, click **Computer Configuration** > **Windows Settings** > **Security Settings** > **Network List Manager Policies**.

13. Right-click **Unidentified Networks** and click **Properties.**

14. Select the **Location type** to **Private** and click **Ok**.

Configuring SOAR

Use the format *username|Computer name* as WinRM credentials. For example, *localadmin|DEV-EXCHANGE18*.

Configuring Domain-Controller for WinRM HTTPS Transport

1. Open the Certificate Authority management console.

2. Right-click **Certificate Templates** and select **Manage**.

3. In the template management console, scroll down and select **Web Server template**.

4. Right-click **Web Server Template**, select **Duplicate Template**.

5. In the **Certificate Property Window** for the new template, navigate to the **General Tab**.

6. Set **Display Name** and **Template Name** to **SOARWINRMHTTPS**.

   Note: Use the same name without spaces. If there is a space that leads to a bug where the process to enroll a new certificate repeats.

7. In the **Subject Name** tab, select **Build from this Active Directory information**.

8. In the **Subject name format** select **Common Name** and select **DNS name**.

9. Click **Security** > specify the **Domain Computers** group for the domain. Allow Read, Enroll and Autoenroll and click **OK**.

10. In the **Certificate Authority management console**, right-click **Certificate Templates** and select **New Template**.

11. Double-click **SOARWINRMHTTPS** and close the window.

12. Navigate to **Start** > **Control Panel**.

13. Select **Administrative Tools** and **Group Policy Management**.

14. In the Menu tree, click **Domains** > **[your domain's name]**.

15. Create a batch script for starting WinRM HTTPS Listener named **SoarWinRMSSLStartupScript.ps1**.

16. Copy and paste the following code into **AtarWinRMSSLStartupScript.ps1**:

```
Start-Transcript C:\Scripts\transaction.log
$sysinfo = Get-WmiObject -Class Win32_ComputerSystem
$server = "{0}.{1}" -f $sysinfo.Name, $sysinfo.Domain
$LatestThumb = Invoke-Command -ScriptBlock {
Get-ChildItem -Path Cert:\LocalMachine\My |
where {$_.subject -match "CN=$server"}
Sort-Object -Property NotAfter -Descending |
Select-Object -Last 1 -ExpandProperty Thumbprint
} -ErrorAction Stop
#If HTTPS Listener does not exist create Listener with quick config.Else evaluate
# available certificates ,sort them by expire date , select first thumbprint
$result=(((Get-ChildItem -Path WSMan:\localhost\Listener).keys) -match 'HTTPS')
if($result.Count -eq 0) {
Set-WSManQuickConfig -UseSSL -Force
} else {
Set-WSManInstance -ResourceURI winrm/config/Listener \
-SelectorSet @{Address="*";Transport="HTTPS"} \
-ValueSet @{CertificateThumbprint=$LatestThumb.Thumbprint[1]}
Restart-Service -Force -Name WinRM
}
Stop-Transcript
```

17. Navigate to **Start** > **Control Panel**.

18. Select Administrative Tools > Group Policy Management.

19. Right-click **WinRM-SOAR** and click **Edit**.

20. Click **Computer Configuration** > **Policies** > **Windows Settings** > **Security Settings** > **Public Key Policies**.

21. Double-click **Certificate Services Client - Auto-Enrollment**.

22. Set the **Configuration Model** to **Enable**.

23. Select **Renew expired certificates**, **update pending certificates**, and **remove revoked**certificates.and **Update certificates that use certificate templates**.

24. Click **Ok.**

25. Click **Computer Configuration** > **Policies** > **Windows Settings** > **Scripts.**

26. Double-click **Startup**.

27. In the **PowerShell Scripts**, click **Add**> **Browse** the file **named AtarWinRMSSLStartupScript.ps1**. and click **OK**.

Force Group Policy Update

Use the following PowerShell commands to force a Policy Update as described in the command block:

```
$computers = Get-ADComputer -Filter *
$computers | ForEach-Object -Process {Invoke-GPUpdate -Computer $_.name \
-RandomDelayInMinutes 0 -Force}
```

Additional Notes

The following patch must be applied to the target computer for WinRM to work without an error:

https://support.microsoft.com/en-us/kb/2842230

Support

Contact Information

Phone	A list of phone numbers is available on the Technical Support Page: https://softwaresupport.softwaregrp.com/support-contact-information
Support Web Site	https://softwaresupport.softwaregrp.com/
ArcSight Product Documentation	https://community.softwaregrp.com/t5/ArcSight-Product-Documentation/ct-p/productdocs

Send Documentation Feedback

If you have comments about this document, you can contact the documentation team by email. If an email client is configured on this computer, click the link above and an email window opens with the following information in the subject line:

Feedback on Integration Guides (SOAR 3.2 3.2)

Just add your feedback to the email and click send.

If no email client is available, copy the information above to a new message in a web mail client, and send your feedback to Documentation-Feedback@microfocus.com.

We appreciate your feedback!