Recorded Future and Zscaler

Integration Deployment Guide

Version 1.0
March 2020

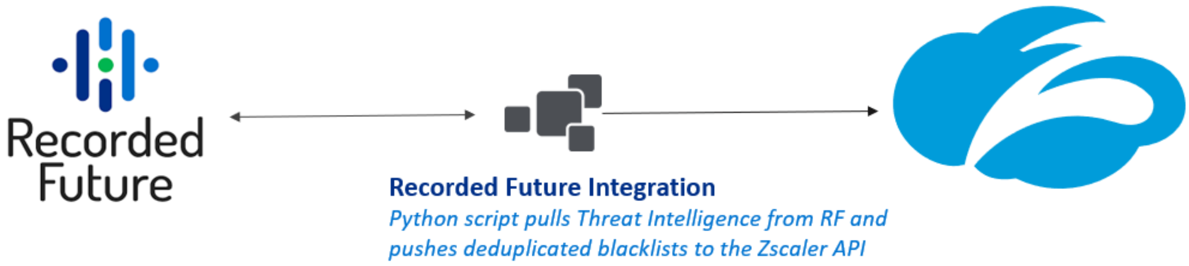
Contents

Introduction

Recorded Future delivers security intelligence to amplify the effectiveness of security and IT teams by informing decisions in real time with contextual, actionable intelligence. Offering a singular view of digital, brand, and third-party risk that's ready for integration, Recorded Future analyzes data from open, closed, proprietary, and aggregated sources.

The Recorded Future and Zscaler integration works by updating the blocklists on a Zscaler customer account. This happens through web API's which connects the two platforms together. The integration is a Python script that can run on a server anywhere that has connectivity between the Recorded Future and Zscaler clouds.

The Integration has been developed and tested by Recorded Future. It is a python script that pulls threat feeds from the Recorded Future platform and calls Zscaler APIs to update blocklists.


Recorded Future
Recorded Future Integration
Python script pulls Threat Intelligence from RF and pushes deduplicated blacklists to the Zscaler API

This Integration was developed using the Recorded Future Connect API & v1 of the Zscaler Cloud API.

Prerequisites

- Zscaler account enabled with API access
- Administrator access to Zscaler to create user and API keys
- Recorded Future API token
- Infrastructure capable of running the Integration Python script

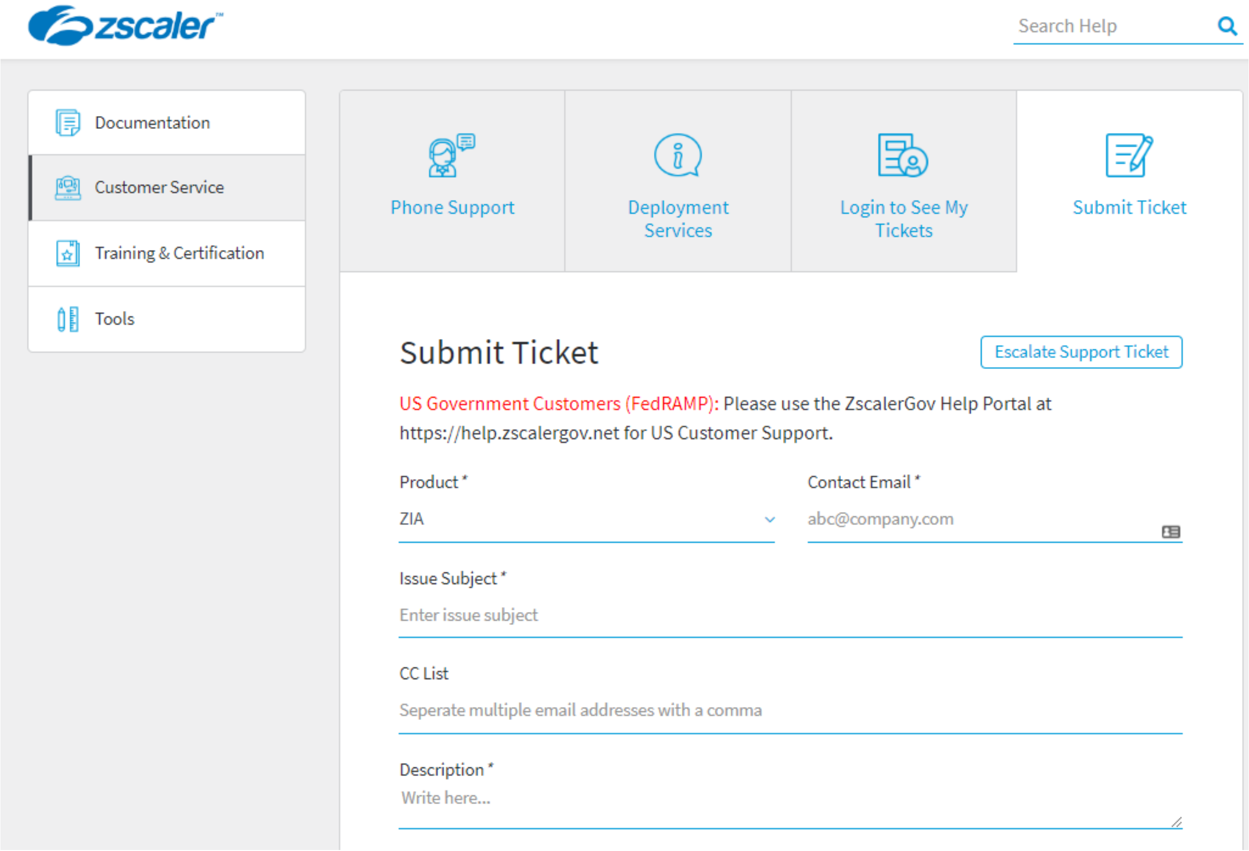
ZIA Configuration

Configuration on the Zscaler side is needed to create a restricted account for API access. At a high level, the steps taken on Zscaler Internet Access (ZIA) interface will include:

- Enabling API access
- Creating an API key
- Creating a limited administrative role for Recorded Future
- Adding an administrator account for Recorded Future
- Committing all the changes

Enabling Zscaler API Access

Start by requesting API access to be added at the Zscaler support portal, https://help.zscaler.com/submit-ticket. Once access has been enabled, you will receive notification and can move to the next step.



Submit Ticket

US Government Customers (FedRAMP): Please use the ZscalerGov Help Portal at https://help.zscalergov.net for US Customer Support.

Product *	Contact Email *
ZIA	abc@company.com

Issue Subject *

Enter issue subject

CC List

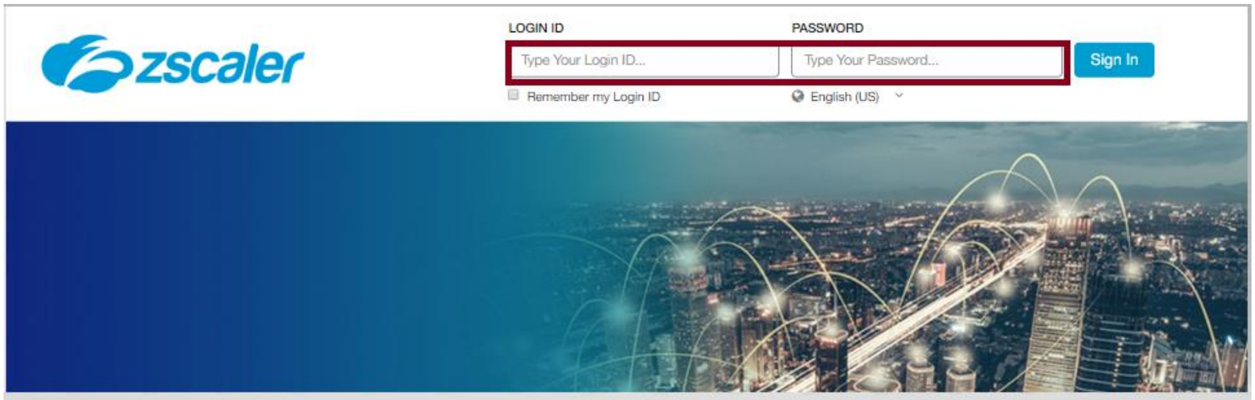
Seperate multiple email addresses with a comma

Description *

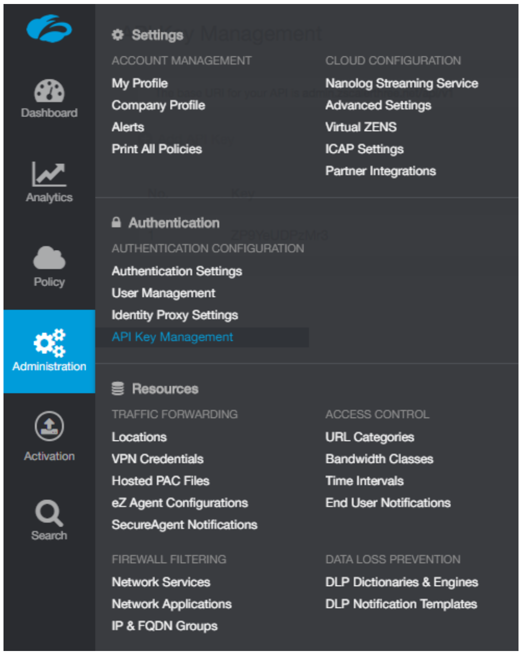
Write here...

Creating a Zscaler API Key

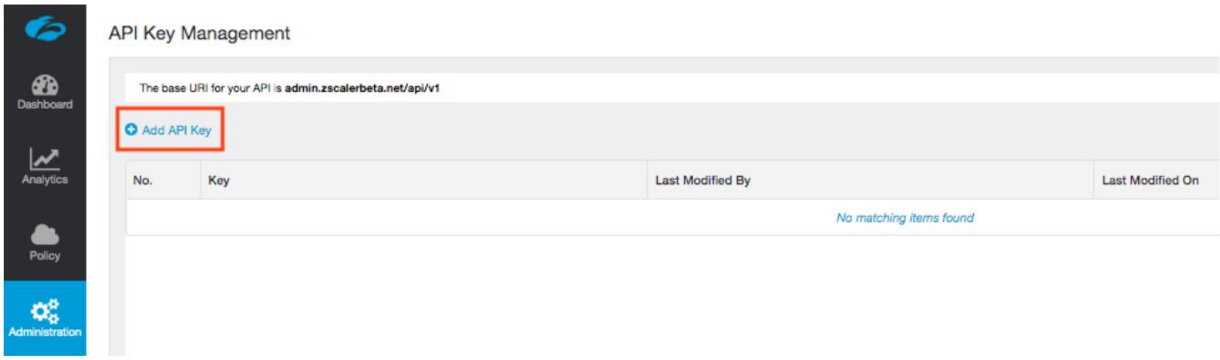
First, we will setup the Zscaler side of this service. Log into Zscaler portal using your administrator account. If you are unable to log in using your administrator account, contact support at https://help.zscaler.com/submit-ticket.



To create an API key, navigate to: Administration > API Key Management



Next, select “**Add API Key**”



API Key Management

The base URI for your API is **admin.zscalerbeta.net/api/v1**

No.	Key	Last Modified By	Last Modified On
		No matching items found	

Once this is performed, the API Key is generated automatically and shown on the screen.

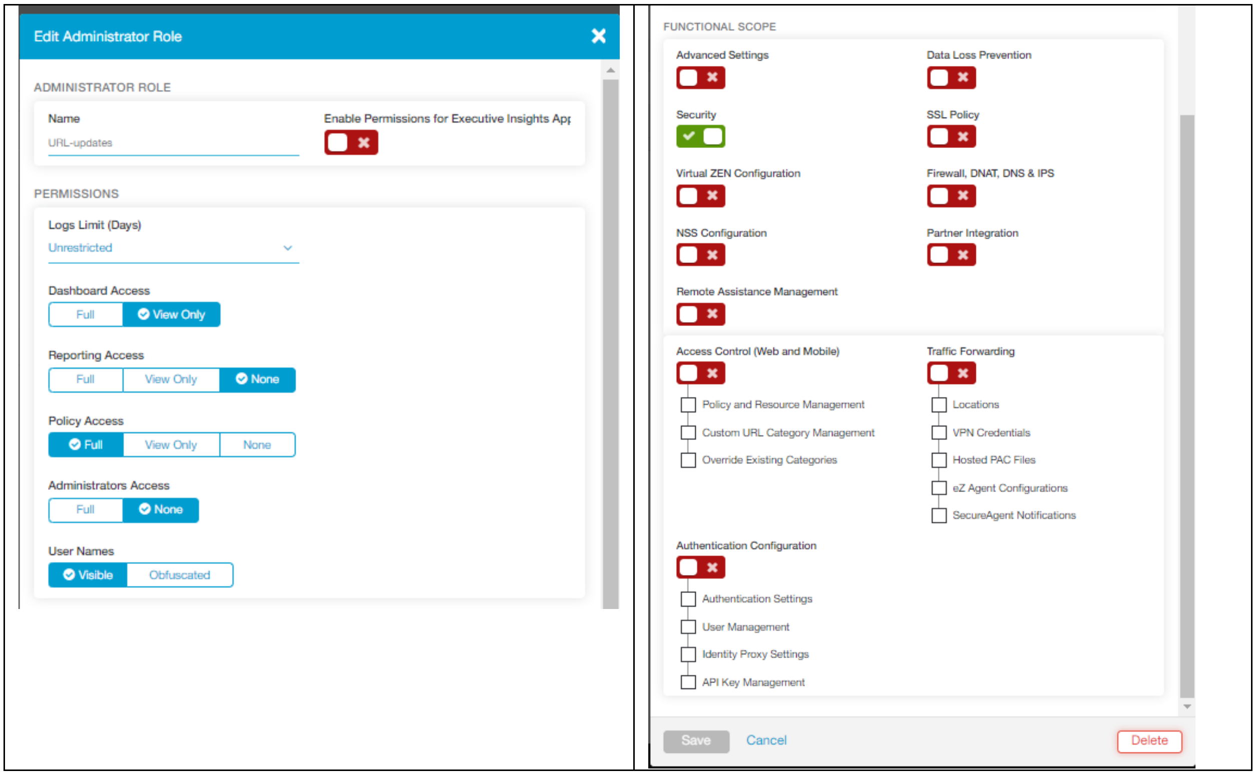
Take note of the base URI and the Key.

Additional details can be found at https://help.zscaler.com/zia/about-api-key-management.

Creating an Administrative Role for Recorded Future

Start by creating a role that will limit permissions to updating the URL blacklists. Navigate to: **Administration -> Role Management -> Add Administrator**.

Create a new role, **URL-updates** is and set the permissions as shown below. Save the changes.



Edit Administrator Role

ADMINISTRATOR ROLE

Name: URL-updates

Enable Permissions for Executive Insights Ap: ☒

PERMISSIONS

Logs Limit (Days): Unrestricted

Dashboard Access: Full [✓ View Only]

Reporting Access: Full View Only [✓ None]

Policy Access: [✓ Full] View Only None

Administrators Access: Full [✓ None]

User Names: [✓ Visible] Obfuscated

FUNCTIONAL SCOPE

Advanced Settings: ☒

Data Loss Prevention: ☒

Security: ✓

SSL Policy: ☒

Virtual ZEN Configuration: ☒

Firewall, DNAT, DNS & IPS: ☒

NSS Configuration: ☒

Partner Integration: ☒

Remote Assistance Management: ☒

Access Control (Web and Mobile): ☒
- □ Policy and Resource Management
- □ Custom URL Category Management
- □ Override Existing Categories

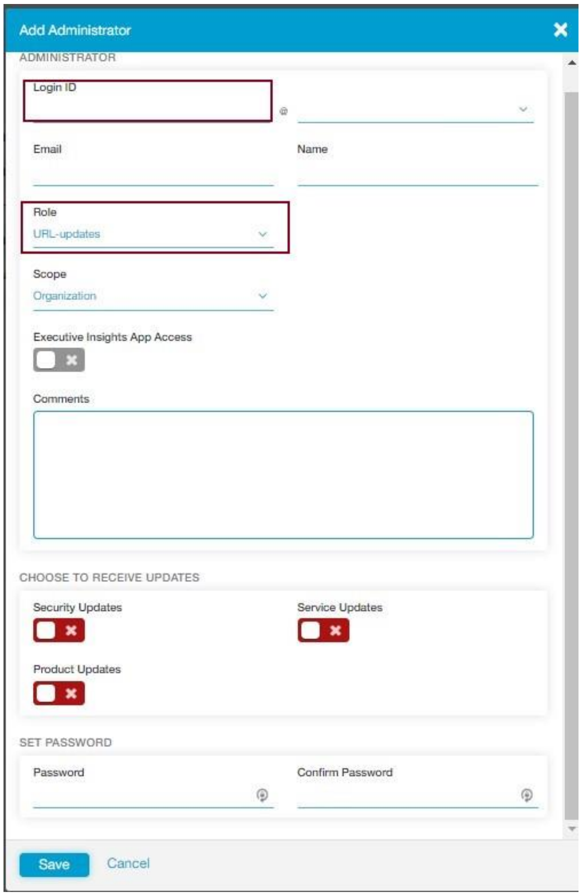
Traffic Forwarding: ☒
- □ Locations
- □ VPN Credentials
- □ Hosted PAC Files
- □ eZ Agent Configurations
- □ SecureAgent Notifications

Authentication Configuration: ☒
- □ Authentication Settings
- □ User Management
- □ Identity Proxy Settings
- □ API Key Management

Adding an Administrative Account for Recorded Future

Next, an administrative account needs to be assigned to the newly created role. Navigate to: **Administration -> Administrator Management -> Add Administrator.**

Set the **Login ID** and **Password** fields for this new account. Be sure to select **URL-updates** as the role for this account.



Add Administrator

ADMINISTRATOR

Login ID

Email

Name

Role
URL-updates

Scope
Organization

Executive Insights App Access

Comments

CHOOSE TO RECEIVE UPDATES

Security Updates

Service Updates

Product Updates

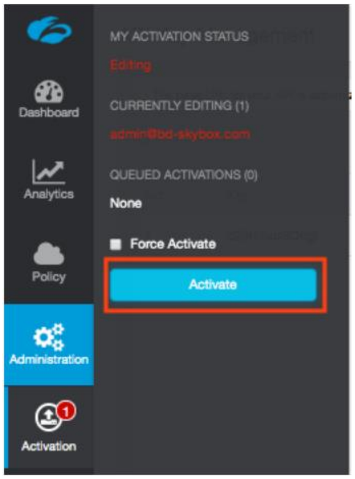
SET PASSWORD

Password

Confirm Password

Committing the Changes

To activate all the changes, go to **Activation** and press the **Activate** button.



Recorded Future Configuration

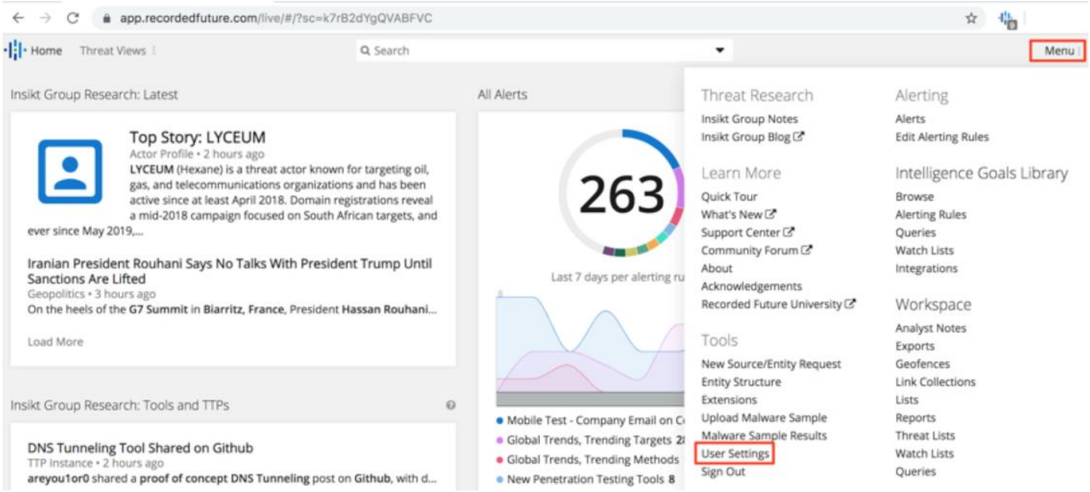
Before using this script, a valid API token is required from Recorded Future.

https://support.recordedfuture.com/hc/en-us/articles/115004179227-Managing-API-tokens

Creating an API Token

Prior to installing the application, it is recommended to configure an API token within the Recorded Future Portal.

1. Login to the Recorded Future Portal (https://app.recordedfuture.com). Click on the menu in the upper right and choose "User Settings".

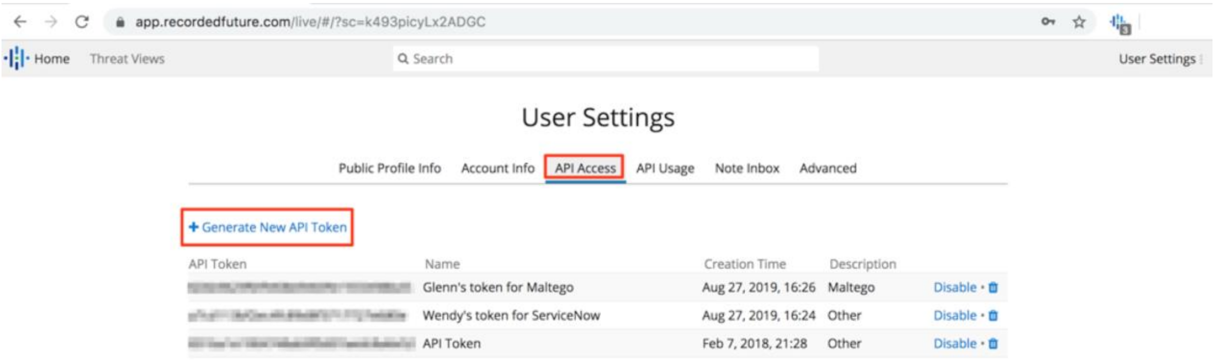


Top Story: LYCEUM
Actor Profile • 2 hours ago
LYCEUM (Hexane) is a threat actor known for targeting oil, gas, and telecommunications organizations and has been active since at least April 2018. Domain registrations reveal a mid-2018 campaign focused on South African targets, and ever since May 2019,...

Iranian President Rouhani Says No Talks With President Trump Until Sanctions Are Lifted
Geopolitics • 3 hours ago
On the heels of the G7 Summit in Biarritz, France, President Hassan Rouhani...

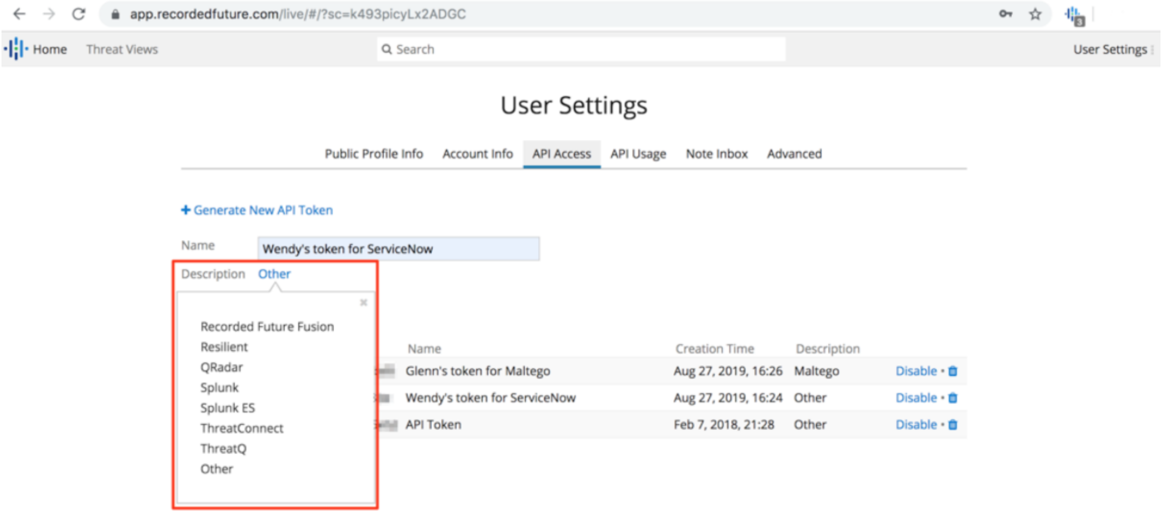
DNS Tunneling Tool Shared on Github
TTP Instance • 2 hours ago
areyou1or0 shared a proof of concept DNS Tunneling post on Github, with d...

2. On the User Settings menu, choose the "API Access" section and click the "+Generate New API Token" link.



API Token	Name	Creation Time	Description	
[redacted]	Glenn's token for Maltego	Aug 27, 2019, 16:26	Maltego	Disable • 🗑
[redacted]	Wendy's token for ServiceNow	Aug 27, 2019, 16:24	Other	Disable • 🗑
[redacted]	API Token	Feb 7, 2018, 21:28	Other	Disable • 🗑

3. Provide a name for your token, select a “Description” of “Other”, and then click the “Create” button. Save the API token that is generated, since you will configure it within the ServiceNow connector for the integration after installation.



Name: Wendy's token for ServiceNow

Description: Other

Name	Creation Time	Description	
Glenn's token for Maltego	Aug 27, 2019, 16:26	Maltego	Disable • 🗑
Wendy's token for ServiceNow	Aug 27, 2019, 16:24	Other	Disable • 🗑
API Token	Feb 7, 2018, 21:28	Other	Disable • 🗑

Deploying the Integration

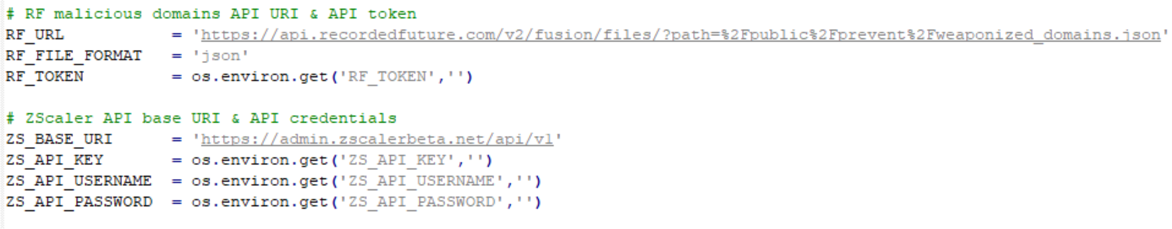
The integration is a Python script and has been tested with Python version 3.6.9. Please contact your Recorded Future representative to get access.

Security Considerations

The Python script should be run in a secure environment. When running the script for the 1st time, additional packages may be downloaded if not already installed on the system. Once the script is running, it is possible to limit the ports to just 443.

Configuration

After obtaining these credentials, the Python script should be updated with the base URI for the Zscaler API and the other credentials. These can also be stored as environment variables. A valid API token is also required from the Recorded Future platform



```
# RF malicious domains API URI & API token
RF_URL            = 'https://api.recordedfuture.com/v2/fusion/files/?path=%2Fpublic%2Fprevent%2Fweaponized_domains.json'
RF_FILE_FORMAT    = 'json'
RF_TOKEN          = os.environ.get('RF_TOKEN','')

# ZScaler API base URI & API credentials
ZS_BASE_URI       = 'https://admin.zscalerbeta.net/api/v1'
ZS_API_KEY        = os.environ.get('ZS_API_KEY','')
ZS_API_USERNAME   = os.environ.get('ZS_API_USERNAME','')
ZS_API_PASSWORD   = os.environ.get('ZS_API_PASSWORD','')
```

The purpose of this script is to download a feed of data from Recorded Future's API and forward the data (a list of malicious domain IoCs) to a cloud Zscaler API instance where a new custom URL category is created and can be used in the creation of ingress/egress blocking policies. This script can be scheduled with a task scheduler to be run on a scheduled basis. The blocked firewall logs ('insights') will show up in the Zscaler instance with whatever custom URL category name is configured in 'ZS_CATEGORY_NAME'.

This script can be used to download and then import domains formatted as a single-column list with no headers with 'RF_FILE_FORMAT' = 'csv', 'json' for a JSON-formatted Security Control Feed.

This script can be used to download and then import domains formatted as a single-column list with no headers with 'RF_FILE_FORMAT' = 'csv', 'json' for a JSON-formatted Security Control Feed.

Before adding the indicators to a new group within Zscaler, all of the Recorded Future indicators are checked against the Zscaler 'urlLookup' API method, in order to determine if the domain would be a duplicate indicator; if so, the indicator is not added. A local cache file can also be used to speed up the identification of domains by keeping a running record of previously validated domains.

Running the Integration

Usage: python zsCustomCategoryImport.py

One point of caution is to ensure that the number of domains being forwarded into the Zscaler instance is below the threshold for the number of URLs which can be concurrently stored between URLs stored in custom categories & the general blacklist. The max limit at the time of this script is 25,000, as both the upper limit for Zscaler and the default limit set in this script. If your Zscaler instance has used custom URLs already, then 'ZS_MAX_DOMAINS' should be adjusted down.

More information on the Zscaler API methods used can be found at https://help.zscaler.com/zia/url-categories-use-cases

Support and Resources

Recorded Future

Support Site:
https://support.recordedfuture.com or email support@recordedfuture.com

Training (Recorded Future University)
https://learning.recordedfuture.com

Zscaler

Zscaler: Getting Started

https://help.zscaler.com/zia/getting-started

Zscaler Knowledge Base:

https://support.zscaler.com/hc/en-us/?filter=documentation

Zscaler Tools:

https://www.zscaler.com/tools

Zscaler Training and Certification:

https://www.zscaler.com/resources/training-certification-overview

Zscaler Submit a Ticket:

https://help.zscaler.com/submit-ticket

ZIA Test Page

http://ip.zscaler.com/