

LetsDefend

Official Incident Report

Event ID: 217

Rule Name: SOC254 - Apache OFBiz Auth Bypass and Code Injection 0-Day
(CVE-2023-51467)

Table of contents

Official Incident Report	1
Event ID: 217	1
Rule Name: SOC254 - Apache OFBiz Auth Bypass and Code Injection 0-Day (CVE-2023-51467)	1
Table of contents	2
Alert	3
Detection	4
Verify	4
Collect Data	5
Analysis	8
Examine The Traffic	11
Containment	17
Summary	18
Lesson Learned	19
Remediation Actions	19
Appendix	20
MITRE ATT&CK	20
Artifacts	21

Alert

Based on the information that the alert provided, it appears that there is a suspicious Web Attack detected on a server named "**Apache OFBiz 16.11.01**" with an IP address of **172.16.17.202**. The Alert is triggered by the **SOC254** rule for **Apache OFBiz Auth Bypass and Code Injection 0-Day (CVE-2023-51467)**.

CVE-2023-51467 vulnerability allows attackers to bypass authentication processes, granting them the ability to remotely execute arbitrary code. The SonicWall threat research team identified this authentication bypass vulnerability during the Root Cause Analysis (RCA) of the previously disclosed CVE-2023-49070 vulnerability.

<https://blog.sonicwall.com/en-us/2023/12/sonicwall-discovers-critical-apache-ofbiz-zero-day-authbiz/>

The device action is marked as "Allowed", indicating that no action was taken by the device to prevent or block the related activities.

#	Severity	Date	Rule Name	EventID	Type	Action
^	High	2024-01-10 1:12	★ SOC254 - Apache OFBiz 0-Day (CVE-2023-51467)	217	Web Attack	
★ SonicWall researchers observed widespread exploitation attempts targeting CVE-2023-51467, identified as a zero-day vulnerability, on December 26, 2023.						
EventID : 217						
Event Time : 2024-01-10 1:12						
Rule : SOC254 - Apache OFBiz 0-Day (CVE-2023-51467)						
Level : Incident Responder						
Hostname : Apache OFBiz 16.11.01						
Destination IP Address : 172.16.17.202						
Source IP Address : 37.19.221.230						
HTTP Request Method : POST						
Requested URL : /webtools/control/xmlrpc/?USERNAME=&PASSWORD=&requirePasswordChange=Y						
User-Agent : python-requests/2.31.0						
Alert Trigger Reason : Anomalous activity detected during a POST request to '/webtools/control/xmlrpc/'.						
L1 Note : The respective device is Ubuntu-based, hosting an Apache OFBiz Docker image. Suspicious network traffic associated with the reported zero-day vulnerability has been identified on the device. Apache OFBiz logs are located within the /ofbiz/runtime/logs directory of the relevant Docker image. Escalating to L2 for an in-depth analysis and investigation.						

The Apache OFBiz server with version 16.11.01 received a POST request from the IP address 37.19.221[.]230. The request was made to the URL '/webtools/control/xmlrpc/' and had no specified username or password. The user agent used for this request was python-requests/2.31.0. This activity was flagged as anomalous, leading to the triggering of an alert.

Based on the provided trigger reason, suspicious activity for CVE-2023-51467 has been detected during a post request on the **Apache OFBiz 16.11.01** which could lead to unauthorized access or manipulation of data.

Detection

Verify

As the playbook suggests we can start investigating the alert by understanding why the alert was triggered

Understand Why the Alert Was Triggered

In order to perform a better analysis and to determine whether the triggered alert is false positive, it is first necessary to understand why the rule was triggered. Instead of starting the analysis directly, first understand why this rule was triggered.

- Examine the rule name. Rule names are usually created specifically for the attack to be detected. By examining the rule name, you can understand which attack you are facing.
- Detect between which two devices the traffic is occurring. It's a good starting point to understand the situation by learning about the direction of traffic, what protocol is used between devices, etc.

Next

Examine the rule name. Rule names are usually created specifically for the attack to be detected. By examining the rule name, you can understand which attack you are facing.

- The above instructions indicate that there has been a flagged anomalous activity involving suspicious activity for CVE-2023-51467 during a post request on the Apache OFBiz 16.11.01. This activity could potentially result in unauthorized access or manipulation of data. By understanding the rule name, it will be possible to determine the nature of the attack being faced.

Detect between which two devices the traffic is occurring. It's a good starting point to understand the situation by learning about the direction of traffic, what protocol is used between devices, etc.

The alert details provide information about the source and destination IP addresses involved in the suspicious network traffic:

- Source IP Address: 37.19.221[.]230
- Destination IP Address (Hostname): 172.16.17.202 (Apache OFBiz 16.11.01)

Collect Data

The next step in the playbook leads us to collect data and gather information about the relevant IP address.

X

Collect Data

Gather some information that can be gathered quickly to get a better understanding of the traffic. These can be summarized as follows.

- Ownership of the IP addresses and devices.
- If the traffic is coming from outside (Internet);
 - Ownership of IP address (Static or Pool Address? Who owns it? Is it web hosting?)
 - Reputation of IP Address (Search in VirusTotal, AbuseIPDB, Cisco Talos)
- If the traffic is coming from company network;
 - Hostname of the device
 - Who owns the device (username)
 - Last user logon time

[Next](#)

Examining whether the IP address or domain has been linked to prior malicious activities and ownership of the IP address can provide insights into the current activity.

Hostname:	Apache OFBiz 16.11.01
IP Address:	172.16.17.202
Version:	OFBiz 16.11.01
Last Logon:	Jan, 11, 2024, 11:32 AM

When going through the technical details in SonicWall's article to check the affected versions, it's noted that Apache OFBiz 16.11.01 with the IP address 172.16.17.202 is affected by this vulnerability.

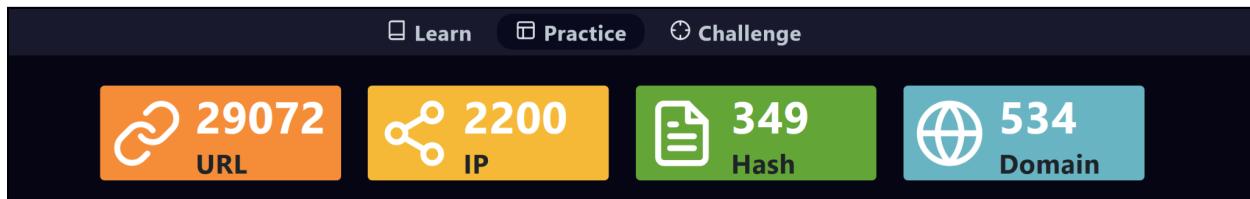
Host Information

Hostname:	Apache OFBiz 16.11.01	Domain:	LetsDefend
IP Address:	172.16.17.202	Bit Level:	64
OS:	Ubuntu 20.04.02	Primary User:	LetsDefend
Client/Server:	Server	Last Login:	Jan, 11, 2024, 11:32 AM

We can check if the traffic is inbound or outbound from the log management system by filtering the IP address of the host. As seen in the log management traffic is inbound.

Jan, 10, 2024, 01:12 PM	Firewall	37.19.221.230	46171	172.16.17.202	8443	
Jan, 10, 2024, 01:27 PM	Firewall	37.19.221.230	59000	172.16.17.202	8443	
Jan, 10, 2024, 01:28 PM	Firewall	37.19.221.230	55605	172.16.17.202	8443	
Jan, 10, 2024, 01:45 PM	Firewall	37.19.221.230	39031	172.16.17.202	8443	
Jan, 10, 2024, 01:48 PM	Firewall	37.19.221.230	24233	172.16.17.202	8443	
Jan, 10, 2024, 01:49 PM	Firewall	37.19.221.230	10664	172.16.17.202	8443	
Jan, 10, 2024, 01:51 PM	Firewall	37.19.221.230	58067	172.16.17.202	8443	

On the LetsDefend threat intel tab, you'll find a comprehensive database dedicated to cataloging maliciously used information, such as IP addresses, domains, and other indicators of compromise.



Upon cross-referencing the destination IP address discovered in the log management system with the Threat Intel tab, it was determined that the address has been categorized as malicious in nature.

The screenshot shows a search interface with various filters. The 'Free text search' field contains '37.19.221.230'. The 'Data type' dropdown is set to 'IP'. The 'Search by data' field also contains '37.19.221.230'. A 'Search' button is visible. Below the filters, a table displays search results with columns: DATE, DATA TYPE, DATA, TAG, and DATA SOURCE. One result row is shown, corresponding to the search term.

DATE	DATA TYPE	DATA	TAG	DATA SOURCE
Jan, 10, 2024, 02:41 PM	IP	37.19.221.230	Malicious	Anonymous

By cross-referencing the IP address with threat intelligence platforms such as Abuseip or Virustotal, we discovered that the IP address is malicious and reported many times.

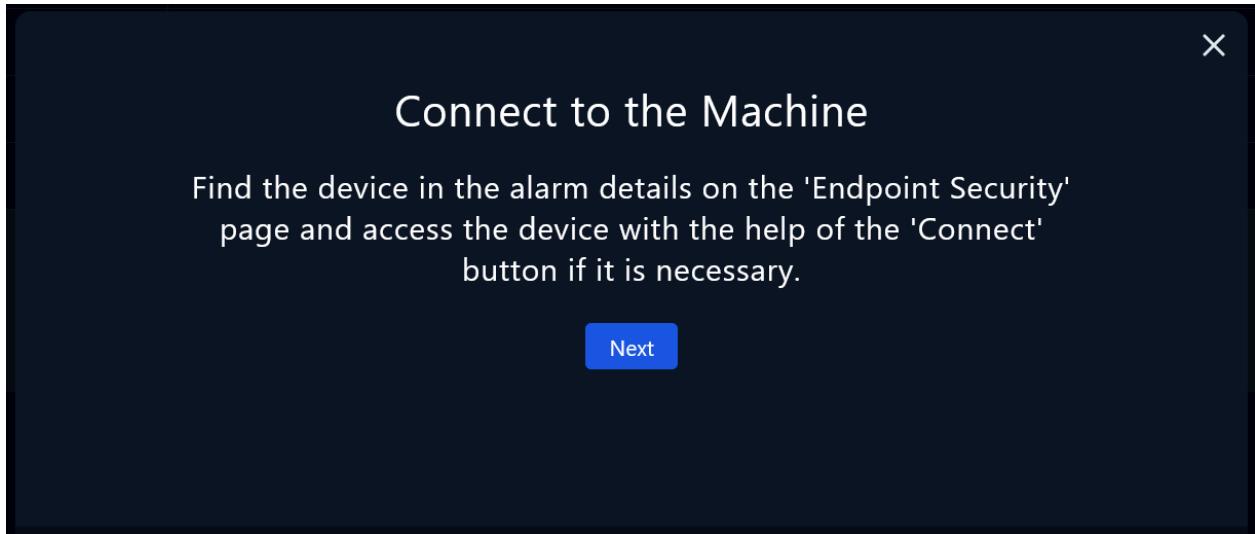
The screenshot shows the VirusTotal analysis interface for the IP address 37.19.221.230. The top bar indicates 2 security vendors flagged it as malicious. Below this, the IP address and its registration information (AS 212238, Datacamp Limited) are displayed. A map shows the location as US. The last analysis date is 1 month ago. The interface includes tabs for DETECTION, DETAILS, RELATIONS, and COMMUNITY (1). Under SECURITY VENDORS' ANALYSIS, it lists results from MalwareURL (Malware), SOC Radar (Malicious), CrowdSec (Not Recommended), and Abusix (Clean). A 'Community Score' section shows a green bar with a value of 1. A 'Do you want to automate checks?' button is visible.

Based on the information provided by VirusTotal, the IP address has been flagged as malicious by **2** antivirus engines. Additionally, in the community tab, it is seen that this IP is contained in a collection.

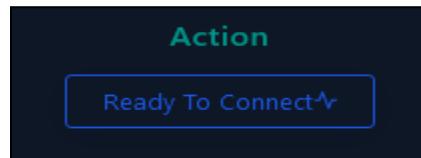
The IOC also seen in the network actions of the host machine.

The screenshot shows a network monitoring interface for the host Apache OFBiz 16.11.01 (172.16.17.202). The host information panel displays details like Hostname, IP Address, OS, and Client/Server status. The action panel shows Containment and Remote Access settings. Below these, a timeline of network events is shown, with several entries for destination IP 37.19.221.230 highlighted with a red box. The timeline includes categories for Processes, Network Action, Terminal History, and Browser History, with a total of 10 results.

Analysis



We can proceed with connecting to the host machine for further analysis. This can easily be done from the Endpoint Security tab by searching for the hostname or IP address and clicking the “Connect” button.



After connecting we can check if the docker is still up. Running this command will allow us to determine the status of the docker.

```
>docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
NAMES					
le4d7de50ed5	marcopinball/ofbiz-demo:latest	/bin/sh -c './gradl...	9 days ago	Up 15 minutes	0.0.0.0:8080->8080/tcp, 0.0.0.0:8443->8443/tcp, 0.0.0.0:8443->8443/tcp

We have identified that the docker is still active and running on the host machine.

The next step is Investigating the access logs. Focusing on IP addresses, user-agents, paths, HTTP status codes and timestamps will help us identify any suspicious or malicious activity.

X

Investigate Access logs

Focus on IP addresses, user agents, URI paths, and HTTP status codes to identify patterns associated with the security incident.

Accessing the Linux Web Server Logs:

- Locate and access the web server logs on the Linux system. Common paths include /var/log/apache2/ for Apache or /var/log/nginx/ for Nginx.

Accessing the Windows Web Server Logs:

- Locate and access web server logs on the Windows system. For IIS, default log paths are often found in %SystemDrive%\inetpub\logs\LogFiles\.

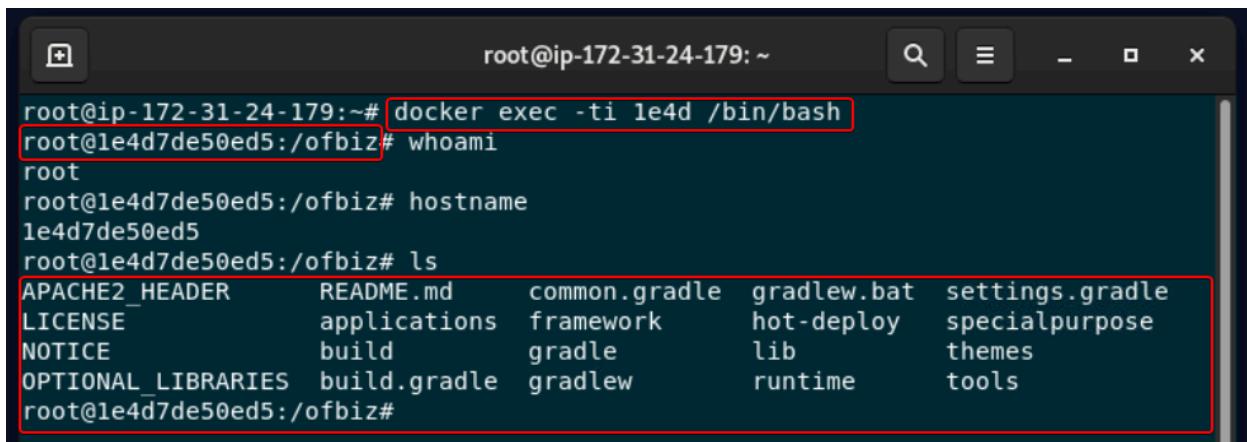
For more detailed information, you can check out the

- Introduction to hacked web server analysis.

[Next](#)

By analyzing these logs, we can gain insights into potential vulnerabilities or security breaches. Additionally, cross-referencing the information with known threat intelligence sources can provide us with a better understanding of the nature and severity of any identified threats.

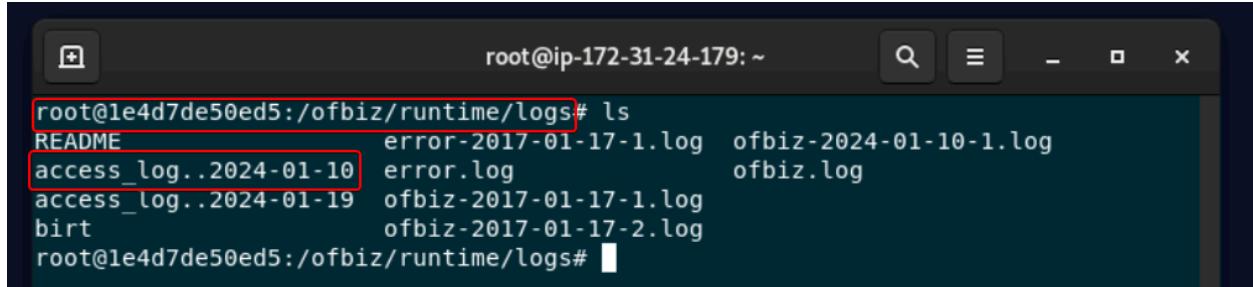
Since the Apache OFBiz running on a docker image we need to gain a shell on the related docker image. To do that we can run the given command:



A terminal window titled 'root@ip-172-31-24-179:~'. The window shows a root shell on a Docker container for Apache OFBiz. The user runs 'docker exec -ti 1e4d /bin/bash' to gain a shell. Inside the shell, they run 'whoami' to confirm they are root. They then run 'hostname' and 'ls' to list the contents of the directory. The output of 'ls' is highlighted with a red box, showing files like APACHE2_HEADER, README.md, common.gradle, gradlew.bat, settings.gradle, LICENSE, applications, framework, hot-deploy, specialpurpose, NOTICE, build, gradle, lib, runtime, themes, OPTIONAL_LIBRARIES, build.gradle, gradlew, and tools.

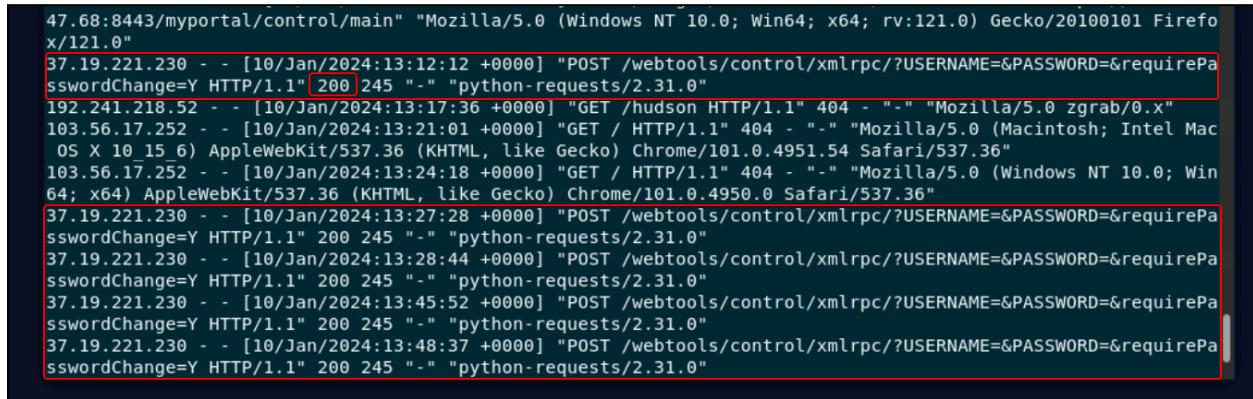
```
root@ip-172-31-24-179:~# docker exec -ti 1e4d /bin/bash
root@1e4d7de50ed5:/ofbiz# whoami
root
root@1e4d7de50ed5:/ofbiz# hostname
1e4d7de50ed5
root@1e4d7de50ed5:/ofbiz# ls
APACHE2_HEADER      README.md      common.gradle  gradlew.bat  settings.gradle
LICENSE            applications   framework     hot-deploy    specialpurpose
NOTICE             build          gradle        lib          themes
OPTIONAL_LIBRARIES build.gradle  gradlew       runtime     tools
root@1e4d7de50ed5:/ofbiz#
```

This analysis can be conducted by examining the logs located within the /ofbiz/runtime/logs directory of the docker image. The directory in question contains the Apache access log for the date when the alarm was triggered.



```
root@1e4d7de50ed5:/ofbiz/runtime/logs# ls
README           error-2017-01-17-1.log  ofbiz-2024-01-10-1.log
access_log..2024-01-10  error.log          ofbiz.log
access_log..2024-01-19  ofbiz-2017-01-17-1.log
birt              ofbiz-2017-01-17-2.log
root@1e4d7de50ed5:/ofbiz/runtime/logs#
```

Upon analyzing the content of the relevant log, it was observed that there are multiple suspicious requests as detailed in the alarm, all originating from the same IP address.



```
47.68:8443/myportal/control/main" "Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:121.0) Gecko/20100101 Firefox/121.0"
37.19.221.230 - - [10/Jan/2024:13:12:12 +0000] "POST /weboots/control/xmlrpc/?USERNAME=&PASSWORD=&requirePa
sswordChange=Y HTTP/1.1" 200 245 "-" "python-requests/2.31.0"
192.241.218.52 - - [10/Jan/2024:13:17:36 +0000] "GET /hudson HTTP/1.1" 404 - "-" "Mozilla/5.0 zgrab/0.x"
103.56.17.252 - - [10/Jan/2024:13:21:01 +0000] "GET / HTTP/1.1" 404 - "-" "Mozilla/5.0 (Macintosh; Intel Mac
OS X 10_15_6) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/101.0.4951.54 Safari/537.36"
103.56.17.252 - - [10/Jan/2024:13:24:18 +0000] "GET / HTTP/1.1" 404 - "-" "Mozilla/5.0 (Windows NT 10.0; Win
64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/101.0.4950.0 Safari/537.36"
37.19.221.230 - - [10/Jan/2024:13:27:28 +0000] "POST /weboots/control/xmlrpc/?USERNAME=&PASSWORD=&requirePa
sswordChange=Y HTTP/1.1" 200 245 "-" "python-requests/2.31.0"
37.19.221.230 - - [10/Jan/2024:13:28:44 +0000] "POST /weboots/control/xmlrpc/?USERNAME=&PASSWORD=&requirePa
sswordChange=Y HTTP/1.1" 200 245 "-" "python-requests/2.31.0"
37.19.221.230 - - [10/Jan/2024:13:45:52 +0000] "POST /weboots/control/xmlrpc/?USERNAME=&PASSWORD=&requirePa
sswordChange=Y HTTP/1.1" 200 245 "-" "python-requests/2.31.0"
37.19.221.230 - - [10/Jan/2024:13:48:37 +0000] "POST /weboots/control/xmlrpc/?USERNAME=&PASSWORD=&requirePa
sswordChange=Y HTTP/1.1" 200 245 "-" "python-requests/2.31.0"
```

- **IP Address:** 37.19.221.230
- **Date and Time:** 10/Jan/2024:13:12:12 +0000
- **HTTP Method:** POST
- **Requested URL:**
/weboots/control/xmlrpc/?USERNAME=&PASSWORD=&requirePasswordChange=Y
- **HTTP Protocol:** HTTP/1.1
- **HTTP Status Code:** 200
- **Bytes Sent:** 245
- **Referrer:** "-"
- **User-Agent:** "python-requests/2.31.0"

This log entry indicates a POST request to the specified URL from the IP address 37.19.221.230, using the user-agent "python-requests/2.31.0", and it received a successful HTTP status code 200 with 245 bytes sent.

Examine The Traffic

The third step of the playbook involves examining the traffic. This step is crucial in identifying any suspicious or malicious activities and understanding the overall network behavior. Additionally, examining the traffic can provide valuable information for further investigation and potential security enhancements.

X

Examine HTTP Traffic

Check the traffic content for any suspicious conditions such as web attack payloads (SQL Injection, XSS, Command Injection, IDOR, RFI/LFI).

Examine all the fields in the HTTP Request. Since the attackers do not only attack through the URL, all the data from the source must be examined to understand whether there is really a cyber attack.

You can review the Web Attacks 101 tutorial for information about attacks on web applications and how to detect these attacks.

- [Web Attacks 101](#)

[Next](#)

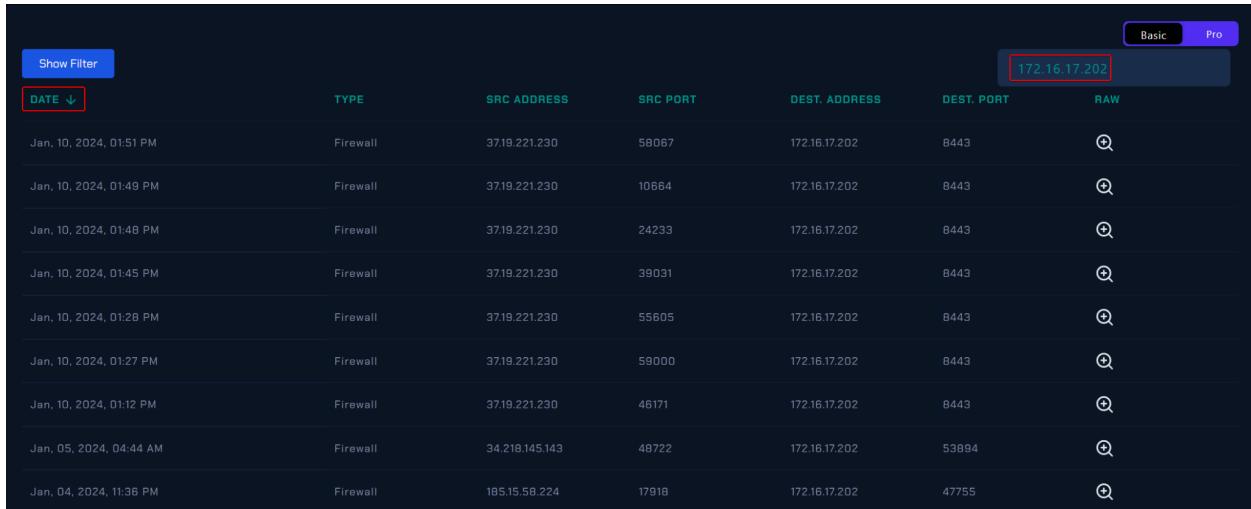
Before examining HTTP traffic, it is crucial to investigate the payloads used in exploiting the relevant vulnerability. The items mentioned in the details of SonicWall's Apache OFBiz security report will facilitate the analysis of the incident.

Testcase 1

Our first test case was based on using empty *USERNAME* and *PASSWORD* parameters while including the parameter *requirePasswordChange=Y* in URI. This test was derived from the testing of CVE-2023-49070 during our signature development to ensure detection in all use cases. The question was posed, what if there is no username and password in the request? For instance, the request might look like

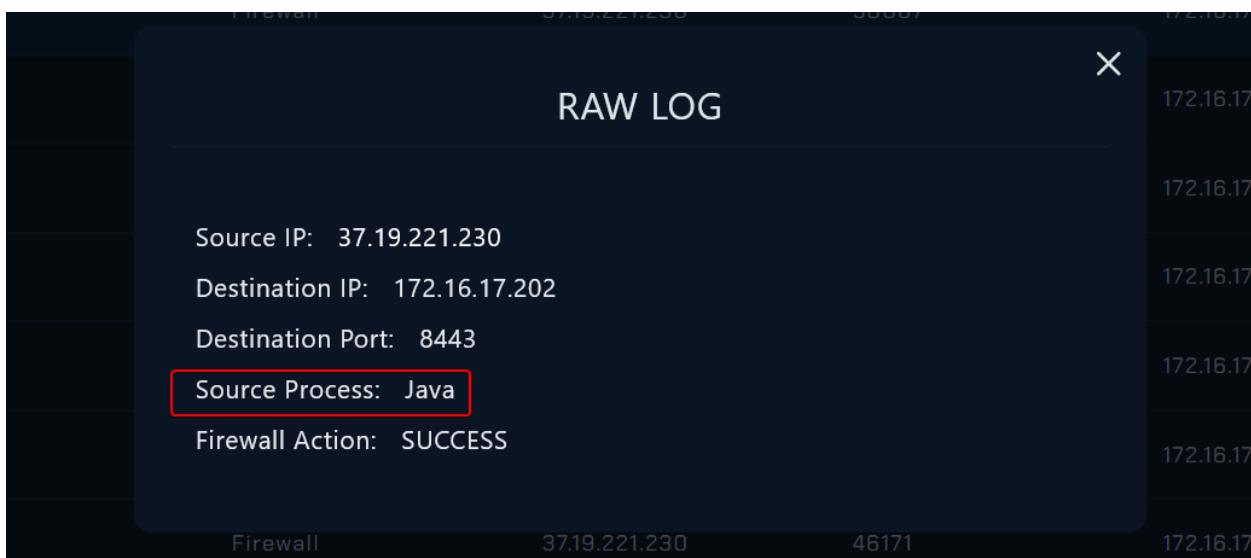
```
https[:]//www.example.com:8443/webtools/control/xmlrpc/?USERNAME=&PASSWORD=&  
requirePasswordChange=Y.
```

Considering that this attack involves a 0-day exploit targeting the Apache OFBiz 16.11.01, we can use the time when the alert was triggered as a reference point for analysis. Filtering the Apache OFBiz 16.11.01 IP address in log management allows us to view the logs.



Show Filter	DATE ↓	TYPE	SRC ADDRESS	SRC PORT	DEST. ADDRESS	DEST. PORT	RAW
	Jan, 10, 2024, 01:51 PM	Firewall	37.19.221.230	58067	172.16.17.202	8443	🔗
	Jan, 10, 2024, 01:49 PM	Firewall	37.19.221.230	10664	172.16.17.202	8443	🔗
	Jan, 10, 2024, 01:48 PM	Firewall	37.19.221.230	24233	172.16.17.202	8443	🔗
	Jan, 10, 2024, 01:45 PM	Firewall	37.19.221.230	39031	172.16.17.202	8443	🔗
	Jan, 10, 2024, 01:28 PM	Firewall	37.19.221.230	55605	172.16.17.202	8443	🔗
	Jan, 10, 2024, 01:27 PM	Firewall	37.19.221.230	59000	172.16.17.202	8443	🔗
	Jan, 10, 2024, 01:12 PM	Firewall	37.19.221.230	46171	172.16.17.202	8443	🔗
	Jan, 05, 2024, 04:44 AM	Firewall	34.218.145.143	48722	172.16.17.202	53894	🔗
	Jan, 04, 2024, 11:36 PM	Firewall	185.15.58.224	17918	172.16.17.202	47755	🔗

Firewall logs for the date of January 10th are available. These logs are essential for monitoring and analyzing network traffic and security events on that specific date.



Source IP:	37.19.221.230	Destination IP:	172.16.17.202	Destination Port:	8443	Source Process:	Java	Firewall Action:	SUCCESS

As seen in the raw log source process for the traffic is Java and the firewall action is success.

Description:
Incident Type:
Created Date:

Is Traffic Malicious?

Decide whether the traffic is malicious or not based on your investigations.

You can find our related training below.

- Web Attacks 101

Malicious Non-malicious

We have observed that the traffic originates from a Java process. Let's proceed to Endpoint Security and analyze the processes. In Endpoint Security, we will identify the specific Java process that is generating the traffic. This will allow us to examine the process's behavior and determine if it is behaving maliciously or if it has been compromised. By analyzing the processes, we can gather more information about the incident and make informed decisions on how to mitigate the threat.

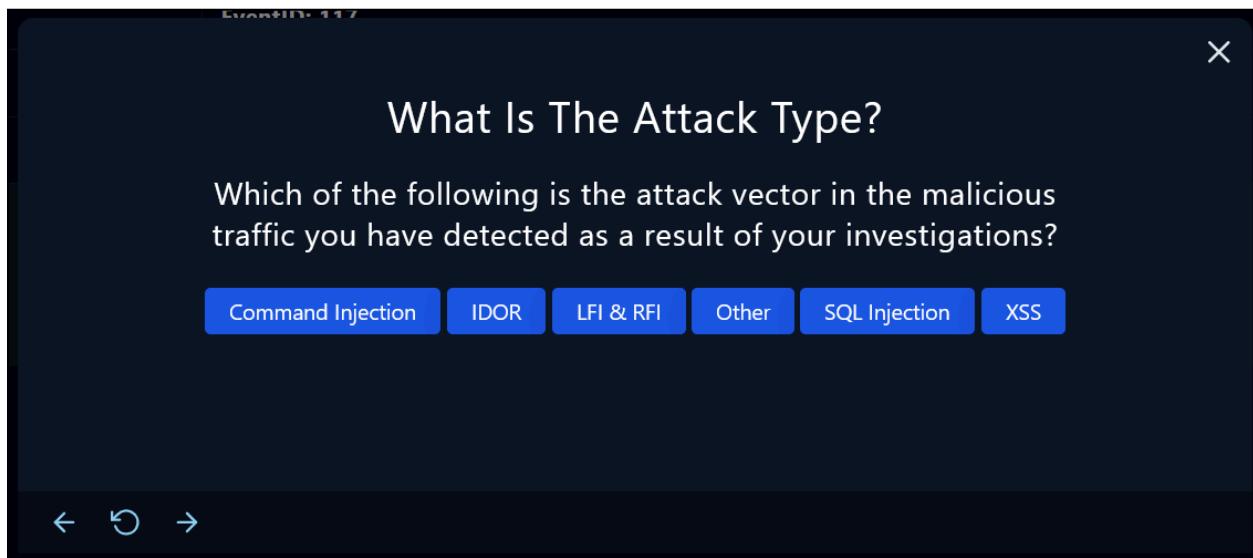
Processes 19 | Network Action 26 | Terminal History 7 | Browser History 1 | Results: 10 ▾

EVENT TIME	PROCESS ID	PROCESS NAME	PARENT PROCESS	COMMAND LINE
Target Process Command Line : <code>cat /etc/passwd</code>				
Image Path : /var/lib/docker/overlay2/74c57cf85146ed5178a3bd06ec74ff8367b038337e26cac5ddda6145df9bb7d/merged/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java				
Process User :				
Parent Name : java				
Parent Path : /var/lib/docker/overlay2/74c57cf85146ed5178a3bd06ec74ff8367b038337e26cac5ddda6145df9bb7d/merged/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java				
Command Line : /usr/lib/jvm/java-8-openjdk-amd64/bin/java -Xms128M -Xmx1024M -Dfile.encoding=UTF-8 -Duser.country -Duser.language=en -Duser.variant -cp /ofbiz/build/libs/ofbiz.jar org.apache.ofbiz.base.start.Start				
2024-01-10 13:49:39.532000	7023	java	java	/usr/lib/jvm/java-8-openjdk-amd64/...
Event Time : 2024-01-10 13:49:39.532000				
Process ID : 7023				
Target Process Command Line : <code>useradd h4xops</code>				
Image Path : /var/lib/docker/overlay2/74c57cf85146ed5178a3bd06ec74ff8367b038337e26cac5ddda6145df9bb7d/merged/usr/lib/jvm/java-8-openjdk-amd64/jre/bin/java				
Process User :				

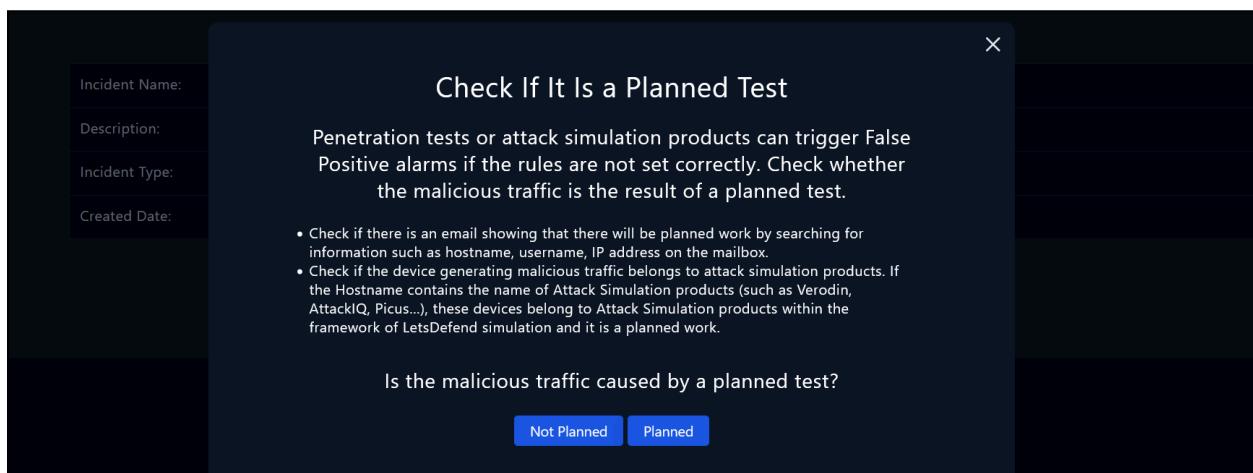
The attacker executed malicious code on the host by sending malicious POST requests.

Based on our analysis, we have confirmed that the traffic is **malicious**.

The next playbook step requires us to find the attack type. The analysis confirms that the relevant attack type is Apache OFBiz Auth Bypass and Code Injection 0-Day (CVE-2023-51467). The answer for the attack type is Other.



When examining the relevant web traffic, it has been observed that the IP address associated with the attacker is listed as an Indicator of Compromise (IOC) in global resources. Furthermore, no evidence suggesting that the respective attack was conducted for testing purposes has been identified in email records or any other section of the investigation.



The IP and hostname information of the relevant hostname were searched within the emails received during the specified dates. However, no evidence related to a planned activity has been observed through this investigation.

The answer for this step is "Not Planned"

The Next step of the playbook involves examining the direction of the traffic.

What Is the Direction of Traffic?

Select the direction of malicious traffic from the available options below.

Format: Source -> Destination

Company Network → Company Network Company Network → Internet
Internet → Company Network

To determine the direction of traffic, we will review the all logs we gathered from our security products on the log management page. The alert creation time will be a key reference for us to investigate the incident.

Jan, 10, 2024, 01:12 PM	Firewall	37.19.221.230	46171	172.16.17.202	8443	
Jan, 10, 2024, 01:27 PM	Firewall	37.19.221.230	59000	172.16.17.202	8443	
Jan, 10, 2024, 01:28 PM	Firewall	37.19.221.230	55605	172.16.17.202	8443	
Jan, 10, 2024, 01:45 PM	Firewall	37.19.221.230	39031	172.16.17.202	8443	
Jan, 10, 2024, 01:48 PM	Firewall	37.19.221.230	24233	172.16.17.202	8443	
Jan, 10, 2024, 01:49 PM	Firewall	37.19.221.230	10864	172.16.17.202	8443	
Jan, 10, 2024, 01:51 PM	Firewall	37.19.221.230	58087	172.16.17.202	8443	

In the log management page, all of the malicious traffic is from the Internet -> Company Network.

What Is the Direction of Traffic?

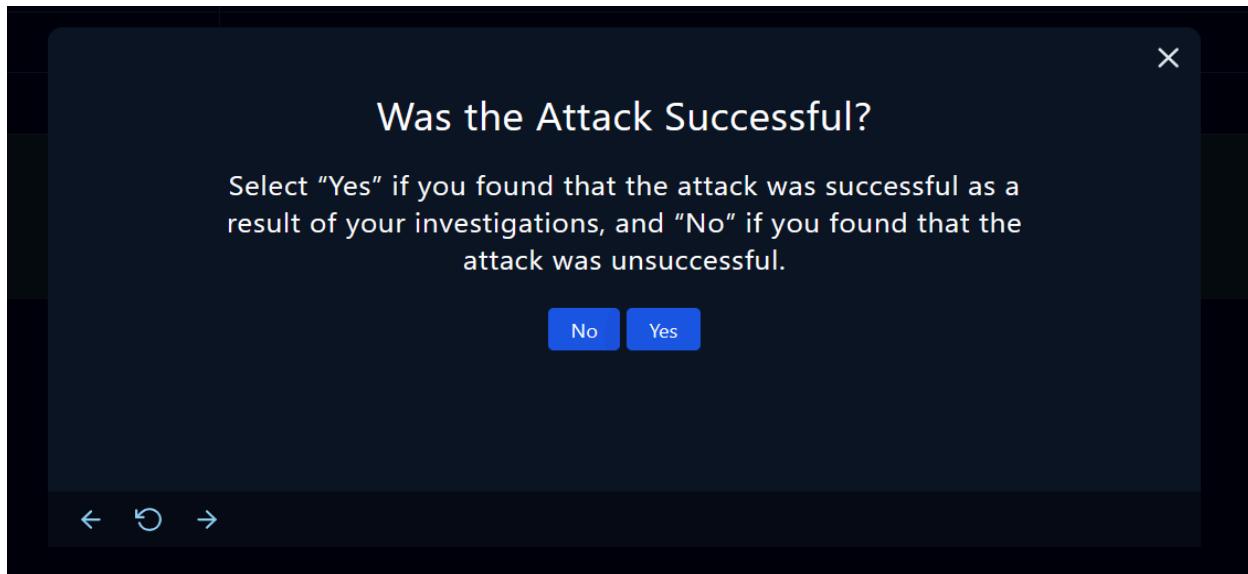
Select the direction of malicious traffic from the available options below.

Format: Source -> Destination

Company Network → Company Network Company Network → Internet
Internet → Company Network

The source address is 37.19.221[.]230 and the destination address is 172.16.17.202. So the answer for this playbook step is Internet -> Company Network.

The next step in the playbook is to assess whether the attack was successful. This involves analyzing the impact of the attacker's actions and determining if they were able to achieve their objectives.



Analyzing the responses enables us to ascertain whether a malicious implant has been detected on the system, thus providing insights into the system's security compromised status.

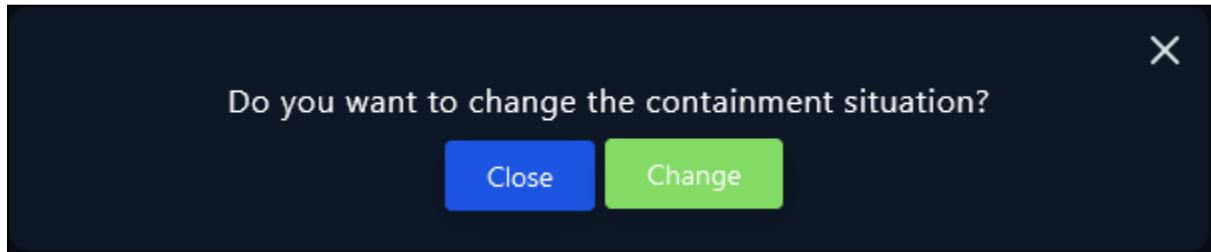
Let's filter the IP address of the machine (172.16.17.202) that initiated these requests on the log management system.

IP Address	Port	Count
172.16.17.234	0	
172.16.17.234	80	
172.16.17.234	80	
172.16.17.234	80	
76.13.32.141	40543	
104.105.45.211	59696	

Based on the HTTP response code of 200, it appears that the request to /server-info.action?bootstrapStatusProvider.applicationConfig.setupComplete=false was successful. Through log analysis, we have confirmed that **the attack was successful**.

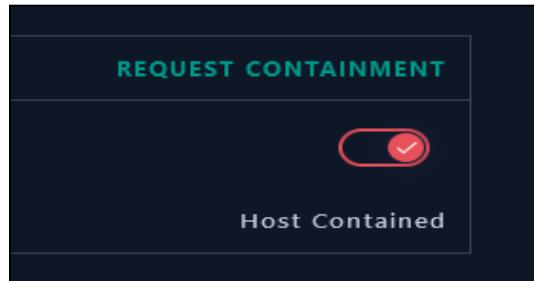
Containment

Based on the information gathered during the investigation, it is highly likely that the system has been compromised. To prevent further data loss or unauthorized access, it is recommended to isolate the system from the network immediately.



Isolation of the host can be made from the endpoint security tab.

Hostname	Apache OFBiz 16.11.01
IP Address	172.16.17.202



After the containment, we can close the alert from the investigation channel.

Summary

The alert report details the detection of suspicious web attack on the Apache OFBiz 16.11.01 (IP: 172.16.17.202) triggered by the SOC235 - Atlassian Apache OFBiz Broken Access Control 0-Day CVE-2023-51467). This vulnerability allows threat actors to gain unauthorized access to sensitive information and perform malicious activities on the affected server.

Upon analysis, it's found that the device action was marked as "allowed," meaning no action was taken to prevent or block the execution of the web attack. The source IP, 37.19.221[.]230, used the user agent "python-requests/2.31.0" to potentially exploit CVE-2023-51467. The investigation, following a detailed playbook, cross-referenced the IP, linking it to malicious activity through threat intelligence platforms like Let's Defend and VirusTotal. SonicWall's report further flagged the IP as Command and Control (C2) and malicious.

Analyzing traffic logs revealed a successful attack pattern, exploiting Apache OFBiz vulnerabilities. Despite an exhaustive examination, no evidence of a planned activity was found in email records. The playbook's final step confirmed the attack's success, as the HTTP response code 200 indicated the completion of the malicious request.

Containment measures are recommended due to the high likelihood of system compromise. Finally, the alert is closed after a thorough investigation, and appropriate actions are taken to mitigate the threat.

Lesson Learned

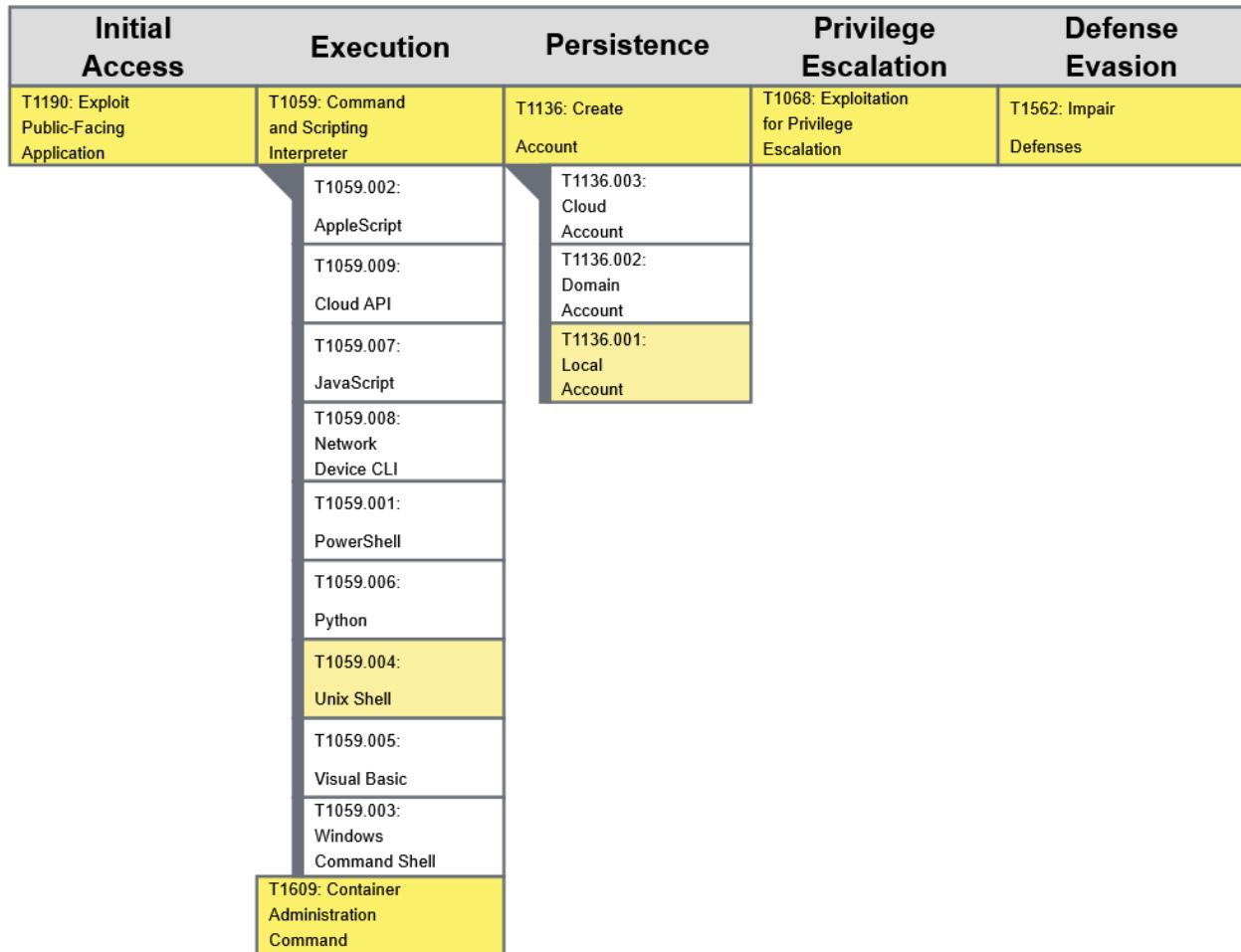
- Timely threat intelligence is crucial for identifying and responding to emerging vulnerabilities and exploits.
- Monitoring for specific indicators of compromise (IOCs) helps detect potential security threats, but they should be supplemented with in-depth analysis.
- Effective threat hunting and detailed investigation are essential to understand the scope of an attack and its potential impact on the organization.
- Staying informed about vulnerabilities and applying patches or mitigations is vital for system security.
- Enabling and collecting logs from various operating systems can significantly enhance visibility into your network's security posture.

Remediation Actions

- Apply security patches or updates to address the CVE-2023-51467 vulnerability in the Apache OFBiz 16.11.01 to eliminate the attack vector.
- Restrict external network access to Apache OFBiz 16.11.01 and Server instances accessible via the public internet, until the necessary upgrades can be performed
- Isolate the compromised machine from the network to prevent the attacker from accessing other resources and systems within the organization.

Appendix

MITRE ATT&CK



MITRE Tactics	MITRE Techniques
Initial Access	T1190: Exploit Public-Facing Application
Execution	T1059: Command and Scripting Interpreter
Execution	T1609: Container Administration Command
Persistence	T1136: Create Account
Privilege Escalation	T1068: Exploitation for Privilege Escalation
Impact	T1562: Impair Defenses

Artifacts

IOC TYPE	VALUE
IPv4	37.19.221[.]230
URI	/xmlrpc/?USERNAME=&PASSWORD=&requirePasswordChange=Y