splunk>

> 深入瞭解 Linux 平臺中的持久性、許可權提升技術和檢測

Splunk 威脅研究團隊添加了 Linux 特權升級和 Linux 持久性技術分析故事，以說明安全運營中心 （SOC） 分析師和安全研究人員在 Linux 操作系統平臺中使用這些技術檢測對手或惡意軟體。在這篇文章中，我們將深入研究這兩種策略的一些流行技術和檢測方法。本文將是我們 2022 年 1 月發佈的文章的深入剖析部分。

分析故事

持久性由不同的技術組成，供攻擊者或惡意軟體作者在啟動、重新啟動計算機甚至憑據更改期間保持其在目標或受感染系統上的立足點和訪問許可權。許可權提升是一種策略，攻擊者試圖為其惡意程式碼獲取提升或更高級別的許可權，以利用 **root** 或**管理員**許可權。這些技術通常與提升上下文中的持久性技術重疊或配合使用。

以下分析是為 Linux 作業系統平台設計的。我們使用 sysmon linux 作為**檢測開發的主要事件日誌集合**。我們建議您閱讀，安裝此工具以及用於此分析的 splunk sysmon TA。

分析案例是我們的威脅研究團隊預構建的檢測和回應所支援的完整安全用例。讓我們討論一下這個分析故事的高級概述，該故事引入了 **32 個新的檢測**。

CRON JOB 和 At Schedule

"At"，cron jobs 也稱為 crontabs，是 UNIX 操作系統上的一個命令行實用程式，用於將作業或任務安排為定期、按固定時間或間隔運行特定腳本或二進位檔。此實用程式通常被攻擊者濫用，以根據其設計的時程表定期執行其惡意程式碼。

技術：

如果已知的 cron jobs 目錄是可寫的，則攻擊者可能會在這些資料夾中放置惡意腳本或二進位檔以自動執行其程式碼。此程式碼將使用 cron jobs 許可權成功執行，從而允許攻擊者提升許可權。

dest	file_create_time	file_name	process_guid	file_path
sysmonlinux-8683	2021-12-15 14:55:22.946	persistevil	{ec23f153-01da-61ba-983a-9298e6550000}	/etc/cron.hourly/persistevil
sysmonlinux-8683	2021-12-15 15:00:55.076	persistevil	{ec23f153-0327-61ba-980a-33b4eb550000}	/etc/cron.hourly/persistevil
sysmonlinux-8683	2021-12-15 15:00:55.103	persistevil	{ec23f153-0327-61ba-985a-83428f550000}	/etc/cron.daily/persistevil
sysmonlinux-8683	2021-12-15 15:00:55.118	persistevil	{ec23f153-0327-61ba-98ea-655926560000}	/etc/cron.weekly/persistevil
sysmonlinux-8683	2021-12-15 15:00:55.128	persistevil	{ec23f153-0327-61ba-985a-9b4a43560000}	/var/spool/cron/crontabs/persistevil

另一種技術是通過在其中附加惡意腳本程式碼來修改正常的 cron jobs 腳本，以隱藏其跟蹤以進行檢測和分析。這可以通過簡單的回顯和粗壯的管道技術來完成，如下面的程式碼所示。

echo "/tmp/evil_cron.sh" >> /etc/cron.daily/logrotate echo "/tmp/evil_cron.sh" >> /etc/cron.hourly/logrotate echo "/tmp/evil_cron.sh" >> /etc/cron.monthly/logrotate echo "/tmp/evil_cron.sh" >> /etc/cron.weekly/logrotate

如果攻擊者已經具有反向 shell 訪問許可權，則還可以通過使用編輯器或使用**「crontab -e」**命令在現有計劃任務中添加 crontab 條目來修改這些正常任務。

"At"調度程序實用程式也是如此。攻擊者可以在 /etc/at.allow 中添加一個條目，該條目是允許執行 「at」 命令的使用者清單。

user	parent_process_name	process_name	process
ubuntu	bash	at	at
ubuntu	bash	at	at 09:00 -f /tmp/evil_work/evil_cron.sh
ubuntu	bash	at	at 09:00 -f /tmp/evil_work/script.sh

檢測：

counts	Name	Technique ID	Tactic	Description
1	Linux Add Files In Known Crontab Directories	T1053.003	Persistence, Privilege Escalation	This analytic detects suspicious file creation (script/binary file) in known cron table directories for persistence or privilege escalation purposes.
2	Linux At Allow Config File Creation	T1053.003	Persistence, Privilege Escalation	This analytic looks for suspicious file creation of /etc/at.allow or /etc/at.deny file. These 2 config files may restrict or allow the user to execute “at” utility tools for scheduling tasks.
3	Linux At Application Execution	T1053.001	Persistence, Privilege Escalation	This analytic identifies a suspicious process creation of “At” application.
4	Linux Edit Cron Table Parameter	T1053.003	Persistence, Privilege Escalation	This analytic identifies a suspicious edit cron jobs using crontab “-e” parameter

5	Linux Possible Append Command To At Allow Config File	T1053.001	Persistence, Privilege Escalation	This analytic looks for suspicious command line that may use to append user entry to /etc/at.allow or /etc/at.deny
6	Linux Possible Append Cronjob Entry on Existing Cronjob File	T1053.003	Persistence, Privilege Escalation	This analytic looks for possible suspicious command lines that may be used to append a code to any existing crontab files for persistence or privilege escalation.
7	Linux Possible Cronjob Modification With Editor	T1053.003	Persistence, Privilege Escalation	This analytic looks for possible modification of cron jobs file using editor.

已知的啟動資料夾

眾所周知，Linux 操作系統有幾個目錄，旨在在啟動或重新啟動期間執行腳本，服務甚至二進位檔。惡意行為者使用這些目錄來持久保存和獲取目標主機的許可權。

技術：

一個例子是 **/etc/init.d** 資料夾。此目錄包含一堆啟動/停止腳本，這些腳本用於在系統運行時或引導期間控制服務守護程式。惡意軟體或對手可能會將惡意腳本丟棄在此資料夾中以實現持久性。

file_create_time	file_name	process_guid	file_path
2021-12-20 15:47:52.565	mal_boot.sh	{ec2c97d1-a5a8-61c0-982a-9a9fe9550000}	/etc/init.d/mal_boot.sh

在 **/etc/profile.d** 資料夾中，包含其他腳本（這些腳本是特定於應用程式的啟動檔，這些腳本也在啟動時由 shell 執行）中的內容相同。

file_create_time	file_name	process_guid	file_path
2021-12-15 12:14:14.644	mal_boot.sh	{ec2fbcb2-dc16-61b9-987a-595b6d550000}	/etc/profile.d/mal_boot.sh

splunk>

檢測:

counts	Name	Technique ID	Tactic	Description
8	Linux File Creation In Init Boot Directory	T1037.004	Persistence, Privilege Escalation	This analytic looks for suspicious file creation on init system directories for automatic execution of script or file upon boot up
9	Linux File Creation In Profile Directory	T1546.004	Persistence, Privilege Escalation	This analytic looks for suspicious file creation in /etc/profile.d directory to automatically execute scripts by shell upon boot up of a linux machine

Linux 服務

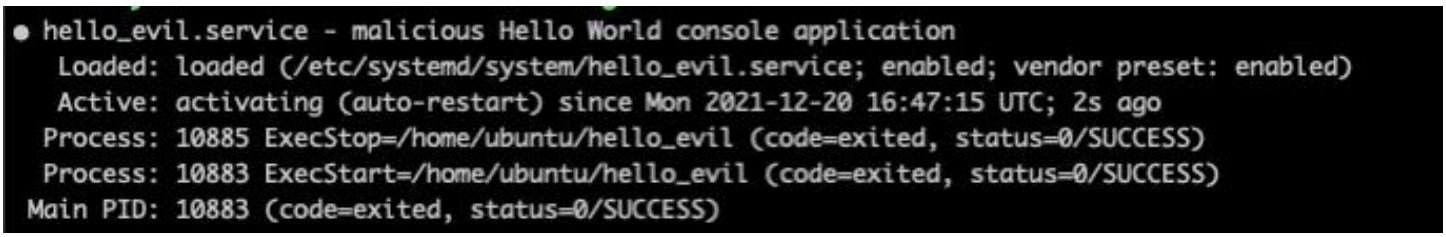
服務是在後台運行的應用程式，等待使用或執行基本任務。一些服務守護程式位於 /etc/init.d 目錄中，而一些位於 /etc/systemd 資料夾中。與 linux 作業系統中的 /etc/init.d 和其他啟動資料夾一樣，此資料夾可能會為對手提供獲得特權提升或持久性的機會。

技術:

例如，刪除指向"/etc/systemd"中的惡意腳本或二進位檔的".service"配置檔可能會為其惡意程式碼創建服務，如下例所示。這種技術也出現在 intezer 分析的 linux golang 惡意軟體中。

[Unit] Description=Hello World console application [Service] # systemd 將運行此可執行檔以啟動服務 ExecStart=/home/ubuntu/hello ExecReload=/home/ubuntu/hello ExecStop=/home/ubuntu/hello Restart=always RestartSec=5 [install] WantedBy=multi-user.target

下面的屏幕截圖顯示了在執行啟動服務命令后如何將".service"檔註冊或激活為服務。



splunk>

有時，攻擊者可能會修改可寫 .service 檔，並對該服務執行 restart/ re-enable 命令，或者停止和啟動服務命令。

user	parent_process_name	process_name	process
root	sudo	systemctl	systemctl enable hello_evil
root	sudo	systemctl	systemctl start hello_evil.service
ubuntu	bash	systemctl	systemctl start hello_evil

user	parent_process_name	process_name	process	process_id
root	sudo	systemctl	systemctl restart hello_evil	10461
ubuntu	bash	sudo	sudo systemctl restart hello_evil	10460

檢測：

counts	Name	Technique ID	Tactic	Description
10	Linux Service File Created In Systemd Directory	T1053.006	Persistence, Privilege Escalation	This analytic looks for suspicious file creation in systemd timer directory in linux platform
11	Linux Service Restarted	T1053.006	Persistence, Privilege Escalation	This analytic looks for restarted or re-enable services in linux platform
12	Linux Service Started Or Enabled	T1053.006	Persistence, Privilege Escalation	This analytic looks for created or enable services in linux platform

添加使用者

攻擊者還可能將使用者添加到受感染的主機以保留並獲得更多特權訪問許可權。以下是用於添加使用者（如 *john*、*atomic_ser2*、*atomicTest*）的常用命令的程式碼示例。

```
sudo useradd -ou 0 -g 0 john

useradd atomic_ser2

sudo adduser atomicTest
```

splunk>

檢測：

counts	Name	Technique ID	Tactic	Description
13	Linux Add User Account	T1136.001	Persistence, Privilege Escalation	This analytic looks for commands to create user accounts on the linux platform.

常見許可權提升技術

就像我們前面提到的，許可權提升策略由不同的技術組成，用於在系統或網路上獲得更高級別的許可權。攻擊者可能會濫用現有的實用程式、配置錯誤甚至漏洞來提升其訪問許可權。在本節中，我們將不包括漏洞利用部分，但我們將處理其他技術來執行此任務。

技術：

攻擊者可以使用 set suid 或 setgid 位執行命令，以便在不同使用者的上下文中運行其程式碼。在 Linux 中，當設置了這些位時，應用程式可以運行提升的許可權上下文。"setcap"實用程式可以像下面的示例一樣設置此位。

```
/usr/bin/setcap cap_net_raw+p /bin/ping
```

或者使用"chmod"實用程式來設置此位。

```
chmod u+s /tmp/evilbin
chmod g+s /tmp/evilbin
```

攻擊者還可以使用「chown」實用程式或「sudo」或「sudo su」來更改文件的擁有權，以獲得升級的訪問許可權。

user	parent_process_name	process_name	process	process_id
root	sudo	chown	chown root evil2_bin	9914
ubuntu	bash	sudo	sudo chown root evil2_bin	9913

另一個可能被濫用於此技術的常見實用程式是"doas"實用程式工具。該工具是作為 sudo 應用程式的簡約替代方案而開發的，sudo 應用程式以提升許可權而聞名。攻擊者可以創建"/etc/doas.conf"，其中包含允許使用者在執行"doas"工具時以 root 身份執行任務的清單。下面是 doas.conf 中的範例設定命令。

```
permit nopass larry cmd reboot
```

用戶可以執行「doas」來運行此配置。

```
doas -C /etc/doas.conf
```

檢測：

counts	Name	Technique ID	Tactic	Description
14	Linux Change File Owner To Root	T1222.002	Persistence, Privilege Escalation	This analytic looks for a command line that change the file owner to root using chown utility tool
15	Linux Setuid Using Chmod Utility	T1548.001	Persistence, Privilege Escalation	This analytic looks for suspicious chmod utility execution to enable SUID bit.
16	Linux Setuid Using Setcap Utility	T1548.001	Persistence, Privilege Escalation	This analytic looks for suspicious setcap utility execution to enable SUID bit.
17	Linux Doas Conf File Creation	T1548.003	Persistence, Privilege Escalation	This analytic is to detect the creation of doas.conf file in linux host platform.
18	Linux Doas Tool Execution	T1548.003	Persistence, Privilege Escalation	This analytic is to detect the doas tool execution in linux host platform.
19	Linux Sudo OR Su Execution	T1548.003	Persistence, Privilege Escalation	This analytic is to detect the execution of sudo or su command in the linux operating system.
20	Linux Common Process For Elevation Control	T1548.001	Persistence, Privilege Escalation	This analytic is to look for possible elevation control access using a common known process in linux platform to change the attribute and file ownership.

splunk>

內核模組載入

與 Windows 操作系統相同，內核模組是一段編譯的二進位程式碼，直接插入內核中或在 ring 0 處插入。此環是最低的，可以訪問系統中的所有內容。複雜的對手和惡意軟體使用這種技術來訪問整個系統，持久化並逃避檢測。

技術：

在 Linux 中，您可以監視已知驅動程式資料夾（"*kernel/drivers/*"）中可疑的內核模組檔創建，如下面的屏幕截圖所示。

file_create_time	file_name	process_guid	file_path
2021-12-22 12:46:56.724	rootkit.ko	{ec2b6afe-1e40-61c3-98aa-156cdd550000}	/lib/modules/5.4.0-1060-aws/kernel/drivers/rootkit/rootkit.ko

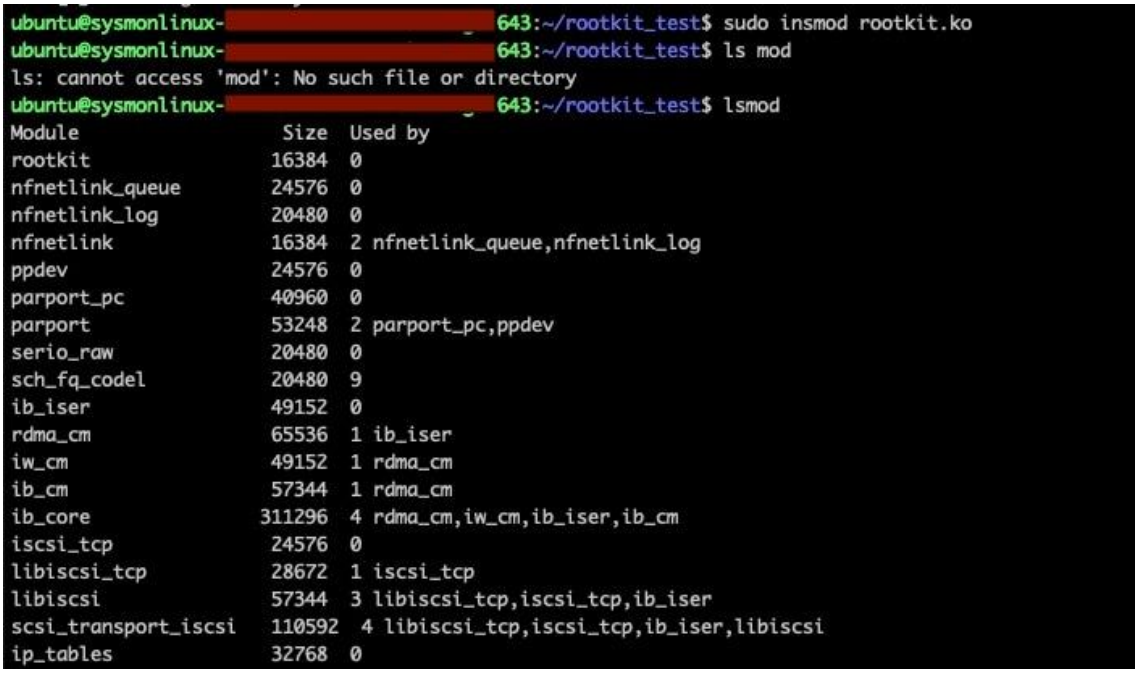
另一種方法是使用「modprobe」或「insmod」將惡意 rootkit 或內核模組載入或插入到內核空間中。程式碼演示如何使用此實用工具。

```
sudo modprobe rootkit.ko
sudo insmod rootkit.ko
```

root	sudo	kmod	modprobe rootkit.ko	22981

user	parent_process_name	process_name	process	process_id
root	sudo	kmod	insmod rootkit.ko	22890
ubuntu	bash	sudo	sudo insmod rootkit.ko	22889

您可以使用「lsmod」實用程式工具列出系統內所有已安裝的內核驅動程式或模組。下面的螢幕截圖顯示了如何驗證「rootkit」內核模組是否插入或安裝在 linux 機器中。



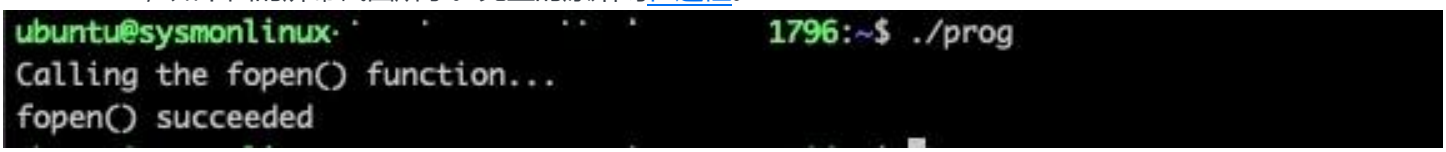
splunk>

檢測：

counts	Name	Technique ID	Tactic	Description
21	Linux File Created In Kernel Driver Directory	T1547.006	Persistence, Privilege Escalation	This analytic looks for suspicious file creation in the kernel/driver directory in the linux platform.
22	Linux Insert Kernel Module Using Insmod Utility	T1547.006	Persistence, Privilege Escalation	This analytic looks for inserting of linux kernel modules using the insmod utility function.
23	Linux Install Kernel Module Using Modprobe Utility	T1547.006	Persistence, Privilege Escalation	This analytic looks for possible installing a linux kernel module using modprobe utility function

劫持庫功能

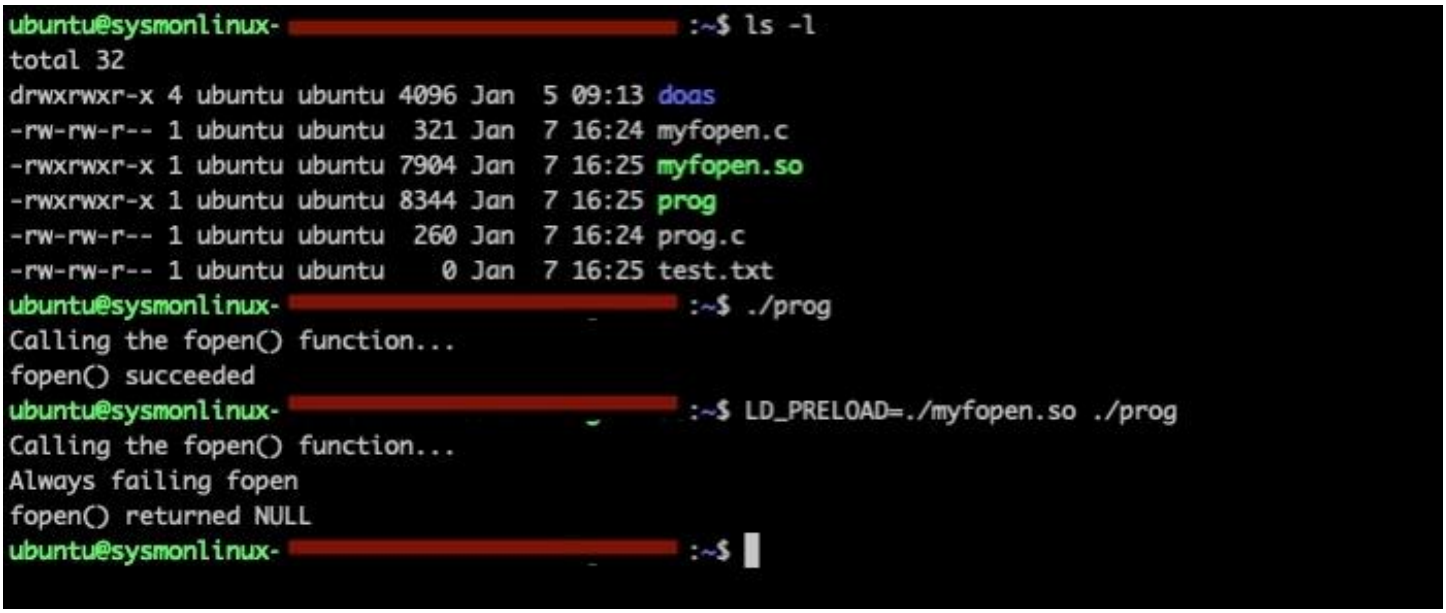
Linux 有一個環境變數，可用於 hijack 或 hook C 標準庫函數。這種技術在幾個對手或惡意軟體中被發現，以執行他們的程式碼。例如，一個簡單的 C 二進位檔，檢查"test.txt"是否存在，如果存在，它將列印"fopen（）suceeded"，如下面的屏幕截圖所示。完整的原始碼在這裡。



但是通過使用 LD_PRELOAD 環境變數和一個".so"模組，該模組將掛接 prog 二進位中的"fopen"函數，以始終列印"fails"，即使檔"test.txt"存在。

myfopen.c 始終列印的模組在挂鉤期間失敗。

LD_PRELOAD 的結果



檢測:

counts	Name	Technique ID	Tactic	Description
24	Linux Preload Hijack Library Calls	T1574.006	Persistence, Privilege Escalation	This analytic is to detect a suspicious command that may hijack a library function using the LD_PRELOAD environment variable in linux platform.

Linux 啟動腳本和憑據檔

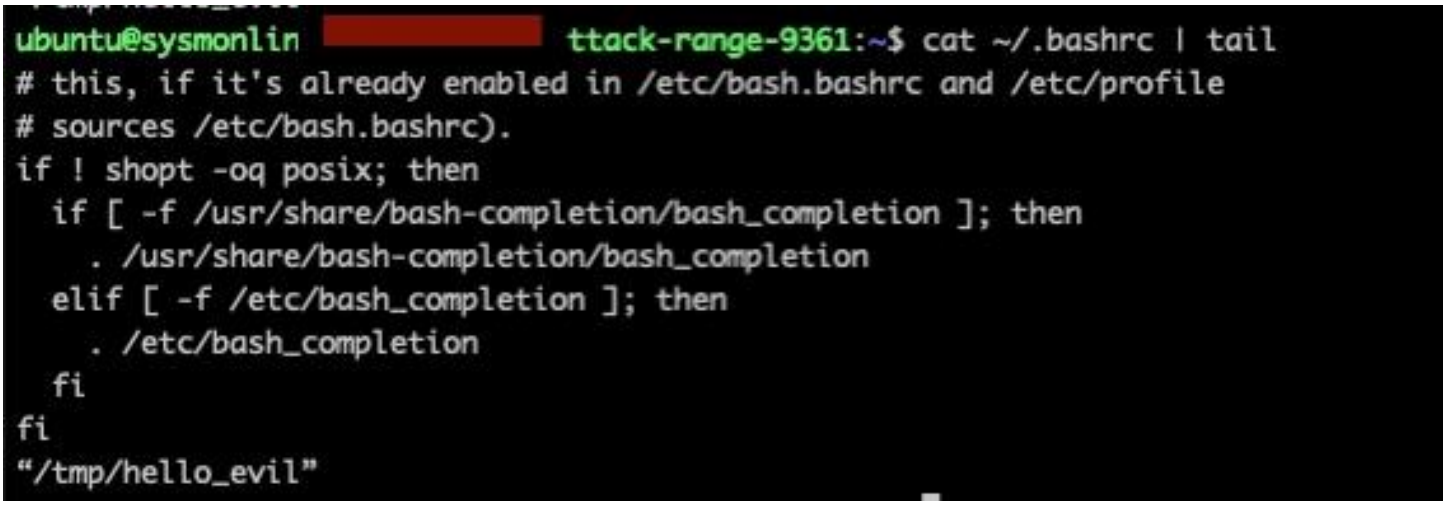
除了 linux 平臺中已知的啟動資料夾之外，還有一些已知的腳本在 linux 電腦重新啟動期間執行。這些腳本還被攻擊者或威脅參與者濫用，以獲得持久性和/或許可權升級到目標系統。

技術:

一個例子是將惡意程式碼附加到其中一個配置檔腳本檔（"*~/.bashrc"，"*~/.bash_profile"，"*/etc/profile"，"~/.bash_login"，"*~/.profile"，"~/.bash_logout"）上，以便在重新啟動計算機時通過 shell 自動執行它。程式碼顯示了使用「echo」 命令和 stdout 管道附加程式碼是多麼簡單。

```
echo “/tmp/hello_evil” >> ~/.bashrc
echo “/tmp/hello_evil” >> /etc/profile
echo “/tmp/hello_evil” >> ~/.bash_profile
```

將結果附加到 ~/.bashrc：



另一種方法是通過訪問或轉儲已知的使用者憑據或控制檔

在目標主機上獲得大量控制。例如，通過將"/etc/sudoers"文件轉儲到另一個包含可以運行命令的控件的檔，存儲使用者資訊和密碼哈希的"/etc/passwd"和"/etc/shadow"檔。下面的程式碼顯示了有權訪問該文件的攻擊者如何將其轉儲到另一個檔以進行破解。

```
sudo cat /etc/shadow > /tmp/shadow_copy
sudo cat /etc/passwd > /tmp/passwd_copy
sudo cat /etc/sudoers > /tmp/sudoers_copy
```

或者通過使用"visudo"或"echo"命令修改"/etc/sudoers"檔來添加控制條目，以執行具有 root 或沒有密碼實例的命令。下面是此攻擊的範例程式碼。

將條目添加到 /etc/sudoers：

```
sudo echo "evil_user ALL=(ALL) NOPASSWD: ALL" >> /etc/sudoers
```

檢測：

counts	Name	Technique ID	Tactic	Description
25	Linux Possible Append Command To Profile Config File	T1546.004	Persistence, Privilege Escalation	This analytic looks for suspicious command lines that are possibly used to modify profile files to automatically execute

				scripts/files by shell upon boot of the machine.
26	Linux Possible Access To Credential Files	T1003.008	Persistence, Privilege Escalation	This analytic is to detect a possible attempt to dump or access the content of /etc/passwd and /etc/shadow to enable offline credential cracking.
27	Linux Possible Access To Sudoers File	T1548.003	Persistence, Privilege Escalation	This analytic is to detect a possible access or modification of /etc/sudoers file.
28	Linux NOPASSWD Entry In Sudoers File	T1548.003	Persistence, Privilege Escalation	This analytic is to look for suspicious command lines that may add entry to /etc/sudoers with NOPASSWD attribute in linux platform.
29	Linux Sudoers Tmp File Creation	T1548.003	Persistence, Privilege Escalation	This analytic is to looks for file creation of sudoers.tmp file cause by editing /etc/sudoers using visudo or editor in linux platform.
30	Linux Visudo Utility Execution	T1548.003	Persistence, Privilege Escalation	This analytic is to looks for suspicious command-line that add entry to /etc/sudoers by using visudo utility tool in linux platform.

SSH Authorized_keys

攻擊者可能會修改 SSH authorized_keys 檔，以保持受害主機上的持久性。威脅參與者還可能修改或訪問" /etc/ssh/sshd_config" 檔，以編輯系統的 SSH 指令 PubkeyAuthentication 和 RSAAuthentication 到特定的 IP 位址或使用者。

counts	Name	Technique ID	Tactic	Description
31	Linux Possible Ssh Key File Creation	T1098.004	Persistence, Privilege Escalation	This analytic is to look for possible ssh key file creation on ~/.ssh/ folder.

32	Linux Possible Access Or Modification Of sshd_config File	T1098.004	Persistence, Privilege Escalation	This analytic is to look for suspicious process command-line that might be accessing or modifying sshd_config.

瞭解更多資訊

您可以在 GitHub 和 Splunkbase 上找到有關安全分析故事的最新內容。Splunk Security Essentials 現在還通過推送更新提供了所有這些檢測。在接下來的幾周里，Splunk 威脅研究團隊將發佈一篇關於這個分析故事的更詳細的文章。敬請期待！

有關安全內容的完整清單，請查看 Splunk Docs 上的發行說明。