Product Solutions Resources Open Source Enterprise Pricing Sign in Sign up

redcanaryco / atomic-red-team Public

Notifications Fork 2.8k Star 9.7k

Code Issues 6 Pull requests 5 Actions Wiki Security Insights

Files

f339e7d

Go to file

- .github
- atomic_red_team
- atomics
  - Indexes
  - T1003.001
  - T1003.002
  - T1003.003
  - T1003.004
  - T1003.005
  - T1003.006
  - T1003.007
  - T1003.008
  - T1003
  - T1006
  - T1007
  - T1010
  - T1012
  - T1014
  - T1016
  - T1018
  - T1020
  - T1021.001
  - T1021.002
  - T1021.003
  - T1021.006
  - T1027.001
  - T1027.002
  - T1027.004
  - T1027
  - T1030
  - T1033
  - T1036.003
  - T1036.004
  - T1036.005
  - T1036.006
  - T1036

atomic-red-team / atomics / T1543.002 / T1543.002.md

CircleCI Atomic Red Team doc... Generate docs from job=gener... 36d49de · 3 years ago History

Preview Code Blame 151 lines (101 loc) · 7.24 KB Raw

T1543.002 - Systemd Service

Description from ATT&CK

> Adversaries may create or modify systemd services to repeatedly execute malicious payloads as part of persistence. The systemd service manager is commonly used for managing background daemon processes (also known as services) and other system resources.(Citation: Linux man-pages: systemd January 2014)(Citation: Freedesktop.org Linux systemd 29SEP2018) Systemd is the default initialization (init) system on many Linux distributions starting with Debian 8, Ubuntu 15.04, CentOS 7, RHEL 7, Fedora 15, and replaces legacy init systems including SysVinit and Upstart while remaining backwards compatible with the aforementioned init systems.
> Systemd utilizes configuration files known as service units to control how services boot and under what conditions. By default, these unit files are stored in the `/etc/systemd/system` and `/usr/lib/systemd/system` directories and have the file extension `.service`. Each service unit file may contain numerous directives that can execute system commands:
>
> - ExecStart, ExecStartPre, and ExecStartPost directives cover execution of commands when a services is started manually by 'systemctl' or on system start if the service is set to automatically start.
> - ExecReload directive covers when a service restarts.
> - ExecStop and ExecStopPost directives cover when a service is stopped or manually by 'systemctl'.
>
> Adversaries have used systemd functionality to establish persistent access to victim systems by creating and/or modifying service unit files that cause systemd to execute malicious commands at system boot.(Citation: Anomali Rocke March 2019)
>
> While adversaries typically require root privileges to create/modify service unit files in the `/etc/systemd/system` and `/usr/lib/systemd/system` directories, low privilege users can create/modify service unit files in directories such as `~/.config/systemd/user/` to achieve user-level persistence.(Citation: Rapid7 Service Persistence 22JUNE2016)

Atomic Tests

- Atomic Test #1 - Create Systemd Service
- Atomic Test #2 - Create Systemd Service file, Enable the service , Modify and Reload the service.

Atomic Test #1 - Create Systemd Service

- T1037.001
- T1037.002
- T1037.004
- T1037.005
- T1039
- T1040

This test creates a Systemd service unit file and enables it as a service.

Supported Platforms: Linux

auto_generated_guid: d9e4f24f-aa67-4c6e-bcbf-85622b697a7c

Inputs:

Name	Description	Type	Default Value
systemd_service_path	Path to systemd service unit file	Path	/etc/systemd/system
systemd_service_file	File name of systemd service unit file	String	art-systemd-service.service
execstoppost_action	ExecStopPost action for Systemd service	String	/bin/touch /tmp/art-systemd-execstoppost-marker
execreload_action	ExecReload action for Systemd service	String	/bin/touch /tmp/art-systemd-execreload-marker
execstart_action	ExecStart action for Systemd service	String	/bin/touch /tmp/art-systemd-execstart-marker
execstop_action	ExecStop action for Systemd service	String	/bin/touch /tmp/art-systemd-execstop-marker
execstartpre_action	ExecStartPre action for Systemd service	String	/bin/touch /tmp/art-systemd-execstartpre-marker
execstartpost_action	ExecStartPost action for Systemd service	String	/bin/touch /tmp/art-systemd-execstartpost-marker

Attack Commands: Run with `bash`!

```
echo "[Unit]" > #{systemd_service_path}/#{systemd_service_file}
echo "Description=Atomic Red Team Systemd Service" >> #{systemd_service_
echo "" >> #{systemd_service_path}/#{systemd_service_file}
echo "[Service]" >> #{systemd_service_path}/#{systemd_service_file}
echo "Type=simple"
echo "ExecStart=#{execstart_action}" >> #{systemd_service_path}/#{system
echo "ExecStartPre=#{execstartpre_action}" >> #{systemd_service_path}/#{
echo "ExecStartPost=#{execstartpost_action}" >> #{systemd_service_path}/
echo "ExecReload=#{execreload_action}" >> #{systemd_service_path}/#{syst
echo "ExecStop=#{execstop_action}" >> #{systemd_service_path}/#{systemd_
echo "ExecStopPost=#{execstoppost_action}" >> #{systemd_service_path}/#{
echo "" >> #{systemd_service_path}/#{systemd_service_file}
echo "[Install]" >> #{systemd_service_path}/#{systemd_service_file}
echo "WantedBy=default.target" >> #{systemd_service_path}/#{systemd_serv
systemctl daemon-reload
systemctl enable #{systemd_service_file}
systemctl start #{systemd_service_file}
```

Cleanup Commands:

```
systemctl stop #{systemd_service_file}
systemctl disable #{systemd_service_file}
rm -rf #{systemd_service_path}/#{systemd_service_file}
systemctl daemon-reload
```

Atomic Test #2 - Create Systemd Service file, Enable the service , Modify and Reload the service.

This test creates a systemd service unit file and enables it to autostart on boot. Once service is created and enabled, it also modifies this same service file showcasing both Creation and Modification of system process.

Supported Platforms: Linux

auto_generated_guid: c35ac4a8-19de-43af-b9f8-755da7e89c89

Attack Commands: Run with `bash`! Elevation Required (e.g. root or admin)

```
cat > /etc/init.d/T1543.002 << EOF
#!/bin/bash
### BEGIN INIT INFO
# Provides : Atomic Test T1543.002
# Required-Start: $all
# Required-Stop :
# Default-Start: 2 3 4 5
# Default-Stop:
# Short Description: Atomic Test for Systemd Service Creation
### END INIT INFO
python3 -c "import os, base64;exec(base64.b64decode('aW1wb3J0IG9zCm9zLnB
EOF

chmod +x /etc/init.d/T1543.002
if [ $(cat /etc/os-release | grep -i ID=ubuntu) ] || [ $(cat /etc/os-rel
systemctl enable T1543.002
systemctl start T1543.002

echo "python3 -c \"import os, base64;exec(base64.b64decode('aW1wb3J0IG9z
systemctl daemon-reload
systemctl restart T1543.002
```

Cleanup Commands:

```
systemctl stop T1543.002
systemctl disable T1543.002
rm -rf /etc/init.d/T1543.002
systemctl daemon-reload
```

Dependencies: Run with `bash`!

Description: System must be Ubuntu ,Kali OR CentOS.

Check Prereq Commands:

```
if [ $(cat /etc/os-release | grep -i ID=ubuntu) ] || [ $(cat /etc/os-rel
```

Get Prereq Commands:

```
echo Please run from Ubuntu ,Kali OR CentOS.
```