elastic

Start free trial

Contact Sales

Platform     Solutions     Customers     Resources     Pricing     Docs

# Privilege Escalation via Named Pipe Impersonation

edit

Identifies a privilege escalation attempt via named pipe impersonation. An adversary may abuse this technique by utilizing a framework such Metasploit's meterpreter getsystem command.

**Rule type**: eql

**Rule indices**:

- winlogbeat-*
- logs-endpoint.events.process-*
- logs-windows.forwarded*
- logs-windows.sysmon_operational-*
- endgame-*
- logs-system.security*
- logs-m365_defender.event-*
- logs-sentinel_one_cloud_funnel.*

**Severity**: high

**Risk score**: 73

**Runs every**: 5m

**Searches indices from**: now-9m ([Date Math format](#), see also `Additional look-back time`)

**Maximum alerts per execution**: 100

**References**:

- https://www.ired.team/offensive-security/privilege-escalation/windows-namedpipes-privilege-escalation
- https://www.cobaltstrike.com/blog/what-happens-when-i-type-getsystem/
- https://redcanary.com/blog/getsystem-offsec/

**Tags**:

- Domain: Endpoint
- OS: Windows
- Use Case: Threat Detection
- Tactic: Privilege Escalation
- Resources: Investigation Guide
- Data Source: Elastic Endgame
- Data Source: Elastic Defend
- Data Source: System
- Data Source: Microsoft Defender for Endpoint
- Data Source: Sysmon
- Data Source: SentinelOne

**Version**: 311

**Rule authors**:

- Elastic

**Rule license**: Elastic License v2

# Investigation guide

**Triage and analysis**

**Investigating Privilege Escalation via Named Pipe Impersonation**

A named pipe is a type of inter-process communication (IPC) mechanism used in operating systems like Windows, which allows two or more processes to communicate with each other by sending and receiving data through a well-known point.

Attackers can abuse named pipes to elevate their privileges by impersonating the security context in which they execute code. Metasploit, for example, creates a service and a random pipe, and then uses the service to connect to the pipe and impersonate the service security context, which is SYSTEM.

**Note**: This investigation guide uses the Osquery Markdown Plugin introduced in Elastic Stack version 8.5.0. Older Elastic Stack versions will display unrendered Markdown in this guide.

**Possible investigation steps**

- Investigate the process execution chain (parent process tree) for unknown processes. Examine their executable files for prevalence, whether they are located in expected locations, and if they are signed with valid digital signatures.
- Investigate other alerts associated with the user/host during the past 48 hours.
- Inspect the host for suspicious or abnormal behavior in the alert timeframe.
- Investigate any abnormal behavior by the subject process such as network connections, registry or file modifications, and any spawned child processes.
- Examine the host for derived artifacts that indicate suspicious activities:

- If any suspicious processes were found, examine the process executable using a private sandboxed analysis system.
- Observe and collect information about the following activities in both the sandbox and the alert subject host:
- Attempts to contact external domains and addresses.
- Use the Elastic Defend network events to determine domains and addresses contacted by the subject process by filtering by the process' `process.entity_id`.
- Examine the DNS cache for suspicious or anomalous entries.
- !{osquery{"label":"Osquery - Retrieve DNS Cache","query":"SELECT * FROM dns_cache"}}
- Use the Elastic Defend registry events to examine registry keys accessed, modified, or created by the related processes in the process tree.
- Examine the host services for suspicious or anomalous entries.
- !{osquery{"label":"Osquery - Retrieve All Services","query":"SELECT description, display_name, name, path, pid, service_type, start_type, status, user_account FROM services"}}
- !{osquery{"label":"Osquery - Retrieve Services Running on User Accounts","query":"SELECT description, display_name, name, path, pid, service_type, start_type, status, user_account FROM services WHERE\nNOT (user_account LIKE *%LocalSystem* OR user_account LIKE *%LocalService* OR user_account LIKE *%NetworkService* OR\nuser_account == null)\n"}}
- !{osquery{"label":"Osquery - Retrieve Service Unsigned Executables with Virustotal Link","query":"SELECT concat(*https://www.virustotal.com/gui/file/*, sha1) AS VtLink, name, description, start_type, status, pid,\nservices.path FROM services JOIN authenticode ON services.path = authenticode.path OR services.module_path =\nauthenticode.path JOIN hash ON

services.path = hash.path WHERE authenticode.result !=
*trusted*\n"}}

- Retrieve the files' SHA-256 hash values using the
  PowerShell `Get-FileHash` cmdlet and search for the
  existence and reputation of the hashes in resources like
  VirusTotal, Hybrid-Analysis, CISCO Talos, Any.run, etc.

**False positive analysis**

- This activity is unlikely to happen legitimately. Benign true
  positives (B-TPs) can be added as exceptions if necessary.

**Response and remediation**

- Initiate the incident response process based on the
  outcome of the triage.
- Isolate the involved host to prevent further post-
  compromise behavior.
- If the triage identified malware, search the environment for
  additional compromised hosts.
- Implement temporary network rules, procedures, and
  segmentation to contain the malware.
- Stop suspicious processes.
- Immediately block the identified indicators of compromise
  (IoCs).
- Inspect the affected systems for additional malware
  backdoors like reverse shells, reverse proxies, or droppers
  that attackers could use to reinfect the system.
- Remove and block malicious artifacts identified during
  triage.
- Run a full antimalware scan. This may reveal additional
  artifacts left in the system, persistence mechanisms, and
  malware components.
- Investigate credential exposure on systems compromised
  or used by the attacker to ensure all compromised
  accounts are identified. Reset passwords for these

accounts and other potentially compromised credentials, such as email, business systems, and web services.

- Determine the initial vector abused by the attacker and take action to prevent reinfection through the same vector.
- Using the incident response data, update logging and audit policies to improve the mean time to detect (MTTD) and the mean time to respond (MTTR).
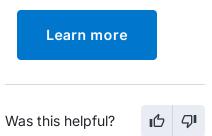
## Rule query

edit

```
process where host.os.type == "windows" and event.ty
(process.name : ("Cmd.Exe", "PowerShell.EXE") or ?p
process.args : "echo" and process.args : ">" and pr
```

**Framework**: MITRE ATT&CK$^{TM}$

- Tactic:

  - Name: Privilege Escalation
  - ID: TA0004
  - Reference URL:
    https://attack.mitre.org/tactics/TA0004/
- Technique:

  - Name: Access Token Manipulation
  - ID: T1134
  - Reference URL:
    https://attack.mitre.org/techniques/T1134/

**ElasticON events are back!**
Learn about the Elastic Search AI Platform from the experts at our live events.

Learn more

Was this helpful?

elastic

The Search AI Company

Follow us

## About us

About Elastic

Leadership

DE&I

Blog

Newsroom

## Join us

Careers

Career portal

## Investor relations

Investor resources

Governance

Financials

Stock

## EXCELLENCE AWARDS

Previous winners

ElasticON Tour

Become a sponsor

All events

## Partners

Find a partner

Partner login

Request access

Become a partner

## Trust & Security

Trust center

EthicsPoint portal

ECCN report

Ethics email

Page 9 of 9