We and selected third parties use cookies or similar technologies for technical purposes and, with your consent, for other purposes as specified in the cookie policy.

Use the "Accept" button to consent. Use the "Reject" button to continue without accepting.

Learn more and customize

# Reject Accept

# Potential Invoke-Mimikatz PowerShell Script



ElasticON events are

Learn about

the Elastic

Search Al

our live events.

Was this

helpful?

Platform from

the experts at

Learn more

16 91

back!

Mimikatz is a credential dumper capable of obtaining plaintext Windows account logins and passwords, along with many other features that make it useful for testing the security of networks. This rule detects Invoke-Mimikatz PowerShell script and alike.

Rule type: query

#### **Rule indices:**

- winlogbeat-\*
- logs-windows.powershell\*

Severity: high

Risk score: 73

Runs every: 5m

Searches indices from: now-9m (Date Math format, see also Additional look-

Maximum alerts per execution: 100

### References:

back time)

- https://attack.mitre.org/software/S0002/
- https://raw.githubusercontent.com/EmpireProject/Empire/master/data/module\_source/credentials/Invokentials/Invokentials.ps1
- https://www.elastic.co/security-labs/detect-credential-access

### Tags:

Domain: Endpoint

• OS: Windows

Use Case: Threat Detection

Tactic: Credential Access

Resources: Investigation Guide

• Data Source: PowerShell Logs

Version: 210

### Rule authors:

Elastic

Rule license: Elastic License v2

# Investigation guide

edit

Triage and analysis

We and selected third parties use cookies or similar technologies for technical purposes and, with your consent, for other purposes as specified in the cookie policy.

Use the "Accept" button to consent. Use the "Reject" button to continue without accepting.

Mimikataz, which are used to dump credentials from the Local Security Authority Subsystem Service (LSASS). Any activity triggered from this rule should be treated with high priority as it typically represents an active adversary.

More information about Mimikatz components and how to detect/prevent them can be found on ADSecurity.

#### Possible investigation steps

- Examine the script content that triggered the detection; look for suspicious DLL imports, collection or exfiltration capabilities, suspicious functions, encoded or compressed data, and other potentially malicious characteristics.
- Investigate the script 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.
- Examine file or network events from the involved PowerShell process for suspicious behavior.
- Investigate other alerts associated with the user/host during the past 48 hours.
- Invoke-Mimitakz and alike scripts heavily use other capabilities covered by other detections described in the "Related Rules" section.
- Evaluate whether the user needs to use PowerShell to complete tasks.
- Investigate potentially compromised accounts. Analysts can do this by searching for login events (for example, 4624) to the target host.
- Examine network and security events in the environment to identify potential lateral movement using compromised credentials.

### False positive analysis

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

### **Related rules**

- PowerShell PSReflect Script 56f2e9b5-4803-4e44-a0a4-a52dc79d57fe
- Suspicious .NET Reflection via PowerShell e26f042e-c590-4e82-8e05-41e81bd822ad
- PowerShell Suspicious Payload Encoded and Compressed 81fe9dc6a2d7-4192-a2d8-eed98afc766a
- Potential Process Injection via PowerShell 2e29e96a-b67c-455a-afe4de6183431d0d
- Mimikatz Memssp Log File Detected ebb200e8-adf0-43f8-a0bb-4ee5b5d852c6
- Modification of WDigest Security Provider d703a5af-d5b0-43bd-8ddb-7a5d500b7da5

### **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.

We and selected third parties use cookies or similar technologies for technical purposes and, with your consent, for other purposes as specified in the cookie policy.

Use the "Accept" button to consent. Use the "Reject" button to continue without accepting.

- protection or antivirus/EDR tools that provide this capability.
- Run a full antimalware scan. This may reveal additional artifacts left in the system, persistence mechanisms, and malware components.
- 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).

### Setup



#### Setup

The PowerShell Script Block Logging logging policy must be configured (Enable).

Steps to implement the logging policy with Advanced Audit Configuration:

```
Computer Configuration >
Administrative Templates >
Windows PowerShell >
Turn on PowerShell Script Block Logging (Enable)
```

Steps to implement the logging policy via registry:

reg add "hklm\SOFTWARE\Policies\Microsoft\Windows\PowerShell\ScriptB

## Rule query



```
event.category:process and host.os.type:windows and
powershell.file.script_block_text:(
    (DumpCreds and
    DumpCerts) or
    "sekurlsa::logonpasswords" or
    ("crypto::certificates" and
    "CERT_SYSTEM_STORE_LOCAL_MACHINE")
)
```

### Framework: MITRE ATT&CK<sup>TM</sup>

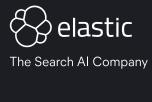
- Tactic:
  - Name: Credential Access
  - ID: TA0006
  - Reference URL: https://attack.mitre.org/tactics/TA0006/
- Technique:
  - Name: OS Credential Dumping
  - ID: T1003
  - Reference URL: https://attack.mitre.org/techniques/T1003/

Page 3 of 5

Sub-technique:

We and selected third parties use cookies or similar technologies for technical purposes and, with your consent, for other purposes as specified in the cookie policy.

Use the "Accept" button to consent. Use the "Reject" button to continue without accepting.



### Follow us











### **About us**

**About Elastic** 

Leadership

DE&I

Blog

Newsroom

### Join us

Careers

Career portal

### **Partners**

Find a partner

Partner login

Request access

Become a partner

# **Trust & Security**

Trust center

EthicsPoint portal

**ECCN** report

Ethics email

### **Investor relations**

Investor resources

Governance

**Financials** 

Stock

# **EXCELLENCE AWARDS**

**Previous winners** 

**ElasticON Tour** 

All events

Potential Invoke-Mimikatz PowerShell Script | Elastic Security Solution [8.15] | Elastic - 02/11/2024 14:42 https://www.elastic.co/guide/en/security/current/potential-invoke-mimikatz-powershell-script.html#potential-invoke-mimikatz-powershell-script

### Notice

We and selected third parties use cookies or similar technologies for technical purposes and, with your consent, for other purposes as specified in the cookie policy.

Use the "Accept" button to consent. Use the "Reject" button to continue without accepting.