initMAX

Wazuh: Threat detection and active protection
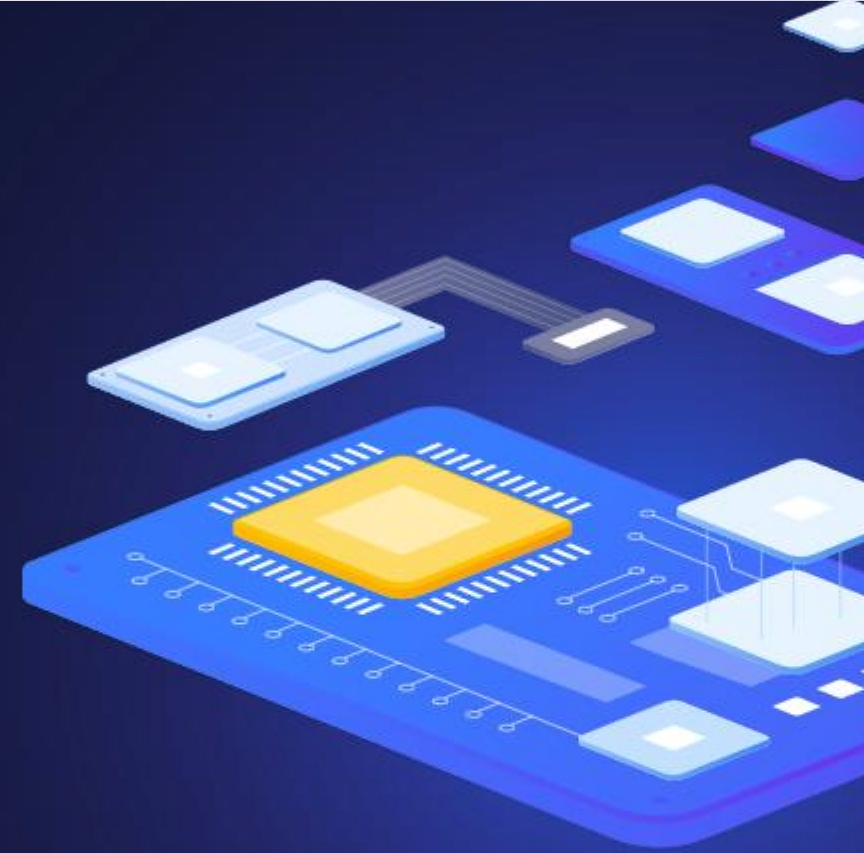
all our microphones are muted

ask your questions in Q&A, not in the Chat

use Chat for discussion, networking or applause

Agenda

1

Intro

2

File Integrity Monitoring (FIM)
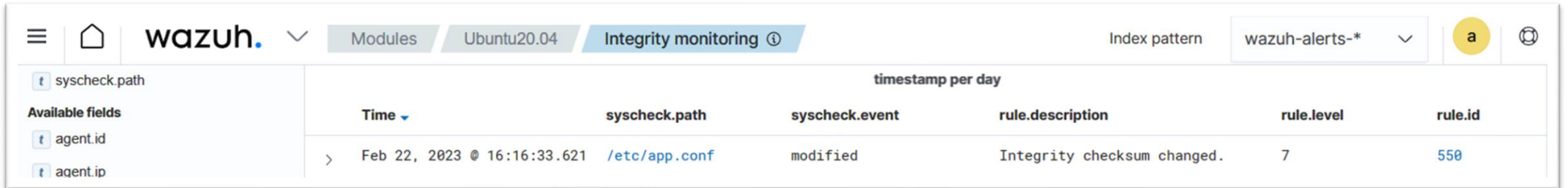
File integrity monitoring (FIM)

❯ Watches selected files or Windows registry and triggers alerts when these files are modified, including changes, additions and deletions

❯ Stores the checksum and other attributes of files

❯ Regularly compares received information against the historical for those files

❯ Supports near real-time file integrity monitoring

❯ [Provides information](#) on who made the changes to the monitored files and the name of the program or process used to make the changes

initMAX

File integrity monitoring (FIM)

File integrity monitoring (FIM)

3

Malware detection with VirusTotal

Malware detection with VirusTotal

› [VirusTotal](#) is an online service that analyzes files and URLs to detect viruses, worms, trojans, and other malicious content using antivirus engines and website scanners

› By sending the hash to the VirusTotal engine, you can know if VirusTotal has already scanned that specific file, and you can analyze its report

› VirusTotal also provides an API that allows access to the information generated by VirusTotal without needing to utilize the HTML website interface

› The VirusTotal public API is limited to 500 requests per day at a rate of 4 requests per minute

› [More informations about VirusTotal API](#)

Malware detection with VirusTotal

› Wazuh FIM looks for any file addition, change, or deletion on the monitored folders

› Integration makes an HTTP POST request to the VirusTotal database using the VirusTotal API.

› This call sends the extracted file hash to compare it with the information in the VirusTotal database

› Integration receives a JSON response

› Wazuh logs the response

› Wazuh integration with external APIs

4

Security Configuration Assessment (SCA)
and custom policies

initMAX

Security Configuration Assessment and custom policies

› Helps maintain a standard configuration through the monitored endpoints

› Use predefined checks based on the Center of Internet Security (CIS)

› Provides periodic scanning and reporting of misconfigurations in the monitored system

› Policies for the SCA scans are written in YAML format

› Policies can be extended or write completely new to fit organization needs

› For example, a rule can be used to look for the existence of a file, a directory, a Windows registry key, or a running process and many others.

› It is also possible to execute a command and check its output against a regular expression

Security Configuration Assessment and custom policies

- Check that a file exists:
  - `f:/path/to/file`
- Check file contents against regex:
  - `f:/path/to/file -> r:REGEX`
- Check if a process is running
  - `p:process_name`
- Check the output of a command
  - `c:command -> output`
- Check the output of a command using regex
  - `c:command -> r:REGEX`
- Check if a registry exists
  - `r:path/to/registry`
- Check if a registry key exists
  - `r:path/to/registry -> key`

initMAX

Security Configuration Assessment and custom policies

- Check for file contents, whole line match:
  - `f:/proc/sys/net/ipv4/ip_forward -> 1`
- Check if a file exists:
  - `f:/proc/sys/net/ipv4/ip_forward`
- Check if a directory contains files:
  - `d:/home -> ^.mysql_history$`
- Check if a directory exists:
  - `d:/etc/mysql`
- Check the running configuration of sshd for the maximum authentication tries allowed:
  - `c:sshd -T -> !r:^\s*maxauthtries\s+4\s*$`
- Check if root is the only account with UID 0:
  - `f:/etc/passwd -> !r:^# && !r:^root: && r:^\w+:\w+:0:`

Demo time

File Integrity Monitoring (FIM)

› Detect creation and modification of cron jobs

› Wazuh by default has a set of rules to detect when changes are made to cron jobs.

› The rules are rules ID 2830, 2831, 2832, 2833, and 2834.

```
<rule id="2832" level="5">
    <if_sid>2830</if_sid>
    <match>REPLACE</match>
    <description>Crontab entry changed.</description>
    <group>pci_dss_10.2.7,pci_dss_10.6.1,gpg13_4.13,gdpr_IV_35.7.d,hipaa_164.312.b,nist_800_53_AU.14,nist_800_53_AU.6,tsc_CC6.8, … </group>
</rule>

<rule id="2833" level="8">
    <if_sid>2832</if_sid>
    <match>REPLACE (root)</match>
    <description>Root's crontab entry changed.</description>
    <mitre>
      <id>T1053.003</id>
    </mitre>
    <group>pci_dss_10.2.7,pci_dss_10.6.1,pci_dss_10.2.2,gpg13_4.13,gdpr_IV_35.7.d,gdpr_IV_32.2,hipaa_164.312.b,nist_800_53_AU.14, …</group>
</rule>
```

File Integrity Monitoring (FIM)

```
### AGENT /var/ossec/etc/ossec.conf  line 110
 <syscheck>
    <directories check_all="yes" realtime="yes" report_changes="yes" >/var/spool/cron/crontabs/</directories>
    <directories check_all="yes" realtime="yes" report_changes="yes" >/etc/crontab</directories>
 </syscheck>

systemctl restart wazuh-agent

### SERVER /var/ossec/etc/rules/local_rules.xml
<group name="initmax_demo,">
  <rule id="100010" level="12">
  <if_sid>550, 554</if_sid>
  <field name="file" type="pcre2">^\/var\/spool\/cron\/crontabs</field>
  <description>Cron job has been modified for user "$(uname)". </description>
  <mitre>
    <id>T1053.003</id>
  </mitre>
</rule>
<rule id="100011" level="12">
  <if_sid>550, 554</if_sid>
  <field name="file" type="pcre2">^\/etc\/crontab</field>
  <description>Crontab file /etc/crontab has been modified. </description>
  <mitre>
    <id>T1053.003</id>
  </mitre>
</rule>
</group>

systemctl restart wazuh-manager
```

Malware detection with VirusTotal

```
### AGENT /var/ossec/etc/ossec.conf line 110

<directories check_all="yes" realtime="yes">/opt/myapp/download/</directories>

# SERVER /var/ossec/etc/ossec.conf before </ossec_config> add

<integration>
  <name>virustotal</name>
  <api_key>6b2d55df126f21bf263874141d</api_key><!-- Replace with your VirusTotal API key -->
  <group>syscheck</group>
  <alert_format>json</alert_format>
</integration>


# test
cd /opt/myapp/download/
curl -LO https://secure.eicar.org/eicar.com && ls -lah eicar.com
```

Custom SCA policies

```
# AGENT

mkdir /var/ossec/etc/custom-sca-files/
touch /var/ossec/etc/custom-sca-files/myapp_check.yml
chown wazuh:wazuh /var/ossec/etc/custom-sca-files/myapp_check.yml
```

Custom SCA policies

```
policy:
  id: "myapp_check"
  file: "myapp_check.yml"
  name: "Wazuh: Detekce hrozeb a aktivní ochrana - demo SCA policy"
  description: "Wazuh: Detekce hrozeb a aktivní ochrana - demo check myapp_check.yml"
  references:
    - https://www.initmax.cz/webinar/wazuh-detekce-hrozeb-a-aktivni-ochrana/
requirements:
  title: "Check that the desired file exists on the monitored endpoints"
  description: "Requirements for running the SCA scans against endpoints with myapp_check.yml on them."
  condition: any
  rules:
    - 'f:/opt/myapp/myapp_config'
checks:
  - id: 10000
    title: "Ensure password is disabled in the test configuration file - FAIL"
    description: "Password is enabled in the test configuration file."
    rationale: "Password is considered weak for the custom test application. Threat actors can brute-force your password."
    remediation: "Disable password by setting the value of the pwd_enabled option to no."
    condition: none
    rules:
      - 'f:/opt/myapp/myapp_config -> r:^pwd_enabled: yes$'
  - id: 10001
    title: "Ensure password is disabled in the test configuration file - PASS"
    description: "Password is enabled in the test configuration file."
    rationale: "Password is considered weak for the custom test application. Threat actors can brute-force your password."
    remediation: "Disable password by setting the value of the pwd_enabled option to no."
    condition: none
    rules:
      - 'f:/opt/myapp/myapp_config -> r:^pwd_enabled: no$'
```

Custom SCA policies
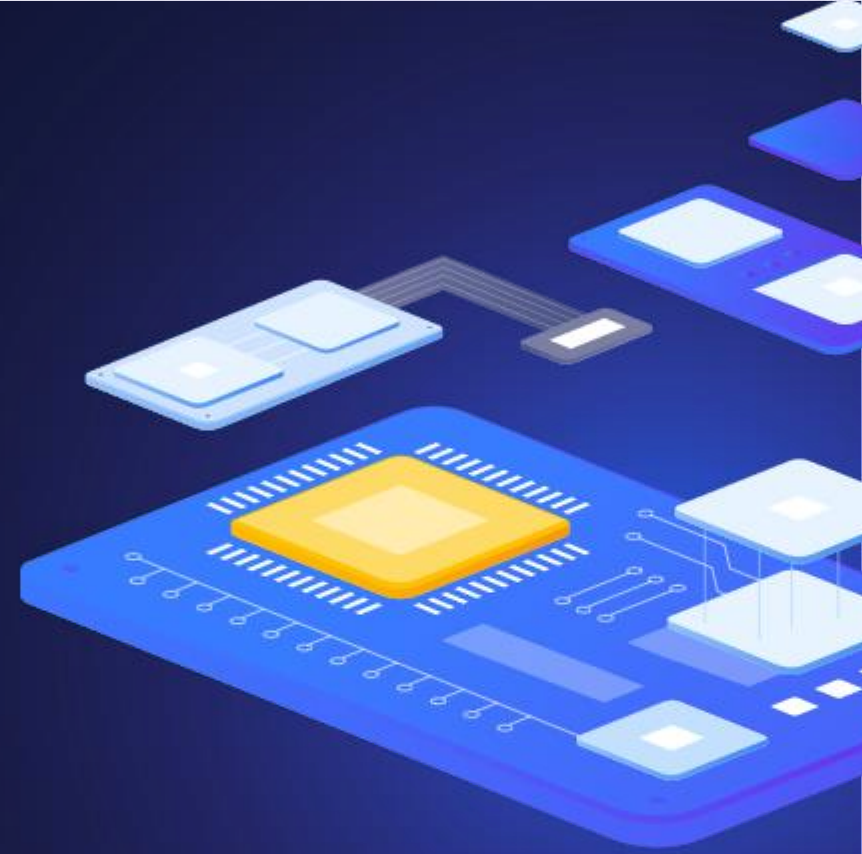
```
# AGENT /var/ossec/etc/ossec.conf before </ossec_config> add
<sca>
  <policies>
    <policy enabled="yes">/var/ossec/etc/custom-sca-files/myapp_check.yml</policy>
  </policies>
</sca>

systemctl restart wazuh-agent
```

initMAX

Questions?

Wazuh: Threat detection and active protection

initMAX

CONTACT US:

Phone:	⟩	+420 800 244 442
Web:	⟩	https://www.initmax.cz
Email:	⟩	tomas.hermanek@initmax.cz
LinkedIn:	⟩	https://www.linkedin.com/company/initmax
Twitter:	⟩	https://twitter.com/initmax
Tomáš Heřmánek:	⟩	+420 732 447 184