CERTITUDE
An open-source and agentless compromise assessment tool
@iansus | @nguvin
$ whoarewe

Jean MARSAULT
CERT-W – Head of R&D
@iansus

» Original dev of CERTitude
» Player for YoloSw4g CTF
» Likes penetration tests and DFIR
» Also likes beer – any suggestion in Amsterdam?

Vincent NGUYEN
Head of the CERT-W
@nguvin

» I’m Batman (shhht)

Internal organization
» Split into 18 business and technological practices
» Working in the Cybersecurity and Digital Trust practice
» Attached to the #Pool-Audit and CERT-W teams

Our activities
» Security audits and penetration tests
» Digital forensics, Incident response and Cyber crises
» Research & Development (5+ ongoing projects)
Historically, the Open Information Security Foundation (OISF) was built circa 2010 around the Suricata IDS/IPS project to support and raise funds for the tool’s development.
Historically, the Open Information Security Foundation (OISF) was built circa 2010 around the Suricata IDS/IPS project to support and raise funds for the tool’s development.

In 2016, the National Cybersecurity Agency of France (ANSSI) joined the OISF as a gold member to highlight the wish to see France get involved in worldwide open-source security projects.
Historically, the Open Information Security Foundation (OISF) was built circa 2010 around the Suricata IDS/IPS project to support and raise funds for the tool’s development.

In 2016, the National Cybersecurity Agency of France (ANSSI) joined the OISF as a gold member to highlight the wish to see France get involved in worldwide open-source security projects.

In 2019, in coordination with the ANSSI, Wavestone decided to present its tool CERTitude as a candidate to be sponsored by the OISF.
CERTitude – the needs behind the tool

Business needs

» A field need: quickly assess which workstations and servers have been compromised

» A compliance need: French companies should be able to perform search campaigns when provided with indicators of compromise (IOC) from the ANSSI

» Also stated in the PRIS qualification from the ANSSI that qualified investigation services providers must be able to perform REC assignments (IOC search)
CERTitude – the needs behind the tool

**Business needs**

- A **field need**: quickly assess which workstations and servers have been compromised.

- A **compliance need**: French companies should be able to perform search campaigns when provided with indicators of compromise (IOC) from the ANSSI.

- Also stated in the **PRIS qualification** from the ANSSI that qualified investigation services providers must be able to perform **REC assignments** (IOC search).

**Infosec incentives**

- Be able to interact with the **worldwide Threat Intelligence sphere**:
  - Be a part of the global "IOC sharing" between private and state actors (MISP, Mitre ATT&CK, etc.)
  - Know your enemy better: “Supreme excellence consists of breaking the enemy's resistance without fighting” (Sun-Tzu)
CERTitude – the needs behind the tool

**Business needs**

» A field need: quickly assess which workstations and servers have been compromised

» A compliance need: French companies should be able to perform search campaigns when provided with indicators of compromise (IOC) from the ANSSI

» Also stated in the PRIS qualification from the ANSSI that qualified investigation services providers must be able to perform REC assignments (IOC search)

**Infosec incentives**

» Be able to interact with the worldwide Threat Intelligence sphere:
  › Be a part of the global “IOC sharing” between private and state actors (MISP, Mitre ATT&CK, etc.)
  › Know your enemy better: “Supreme excellence consists of breaking the enemy's resistance without fighting” (Sun-Tzu)

**Chronology**

2013

» Scanopy, a first approach to automating IOC tooling

2014 / 2015

» First version of CERTitude, presented at the CORI&IN

2015 to 2017

» First tests of the tool internally and with private partners

2019

» Start of the global CERTitude rework – start from scratch
» Candidate tool to be sponsored by the OISF

All-hands-on-deck DFIR crises in July and October introduced delay in the development
A short story about digital forensics assignments

1. The assignments starts after the victim calls, usually after having noticed a suspicious element or behavior in one (or more) asset.
A short story about digital forensics assignments

1. The assignments start after the victim calls, usually after having noticed a suspicious element or behavior in one (or more) asset.

2. You start investigating the asset until you are able of drawing a painting of what the threat looks like: these are indicators of compromise (IOC).
A short story about digital forensics assignments

1 The assignments starts after the victim calls, usually after having noticed a suspicious element or behavior in one (or more) asset.

2 You start investigating the asset until you are able of drawing a painting of what the threat looks like: these are indicators of compromise (IOC).

3 You would normally search for static signatures, but the threat lives memory only. The only solution remains in searching for behavioral indicators on the Information System’s assets.
A short story about digital forensics assignments

1. The assignments start after the victim calls, usually after having noticed a suspicious element or behavior in one (or more) asset.

2. You start investigating the asset until you are able of drawing a painting of what the threat looks like: these are indicators of compromise (IOC).

3. You would normally search for static signatures, but the threat lives memory only. The only solution remains in searching for behavioral indicators on the Information System’s assets.

4. Unfortunately, logs are sparse, have a short retention time and are not centralized. In this moment, you wish you could be able to assess the live presence of behavioral indicators on many assets at a time (e.g. whole Windows domain(s) or forest(s))...
Introducing… CERTitude

CERTitude is …

» An automated collection and analysis tool – data is gathered from the assets and analyzed on a centralization server
» Agentless – you do not need having installed any agent beforehand on the assets
» Discrete – indicators of compromise searched for are not disclosed on the assets
» Respectful of your data – collected data does not leave your information system
Introducing... CERTitude

**CERTitude is ...**

- An automated collection and analysis tool – data is gathered from the assets and analyzed on a centralization server
- Agentless – you do not need having installed any agent beforehand on the assets
- Discrete – indicators of compromise searched for are not disclosed on the assets
- Respectful of your data – collected data does not leave your information system
Introducing... CERTitude

CERTitude is ...

» An automated collection and analysis tool – data is gathered from the assets and analyzed on a centralization server
» Agentless – you do not need having installed any agent beforehand on the assets
» Discrete – indicators of compromise searched for are not disclosed on the assets
» Respectful of your data – collected data does not leave your information system

CERTitude is not ...

» a blind threat hunting tool – it needs feeding, tuning, and manual checks for false positives in results
Under the hood

CERTitude operates with **three different tiers**, as shown below:

---

**CERTitude**

*Master node*
Under the hood

CERTitude operates with **three different tiers**, as shown below:

- **Network zone 1 (US)**
- **Network zone 2 (APAC)**
- **Network zone 3 (EU)**

*CERTitude*
Master node

*Company IS*
Workstation and servers
CERTitude operates with **three different tiers**, as shown below:
Step 1 – Create your hunting environment

Creating users

» You need first to install the master node on a dedicated virtual machine or host

» Then, you can create users with either administrator, user or viewer rights

» Administrators and users will both have the ability to log in on the master and centralizers

Uploading indicators

» Upon the completion on your first seemingly compromised assets, you can build your first indicator of compromise

» When uploaded on the master node, it will get broken down in elementary indicators to facilitate further analyses

Technology sneak peak

» The master node API is written in Python3 with Flask and SQLAlchemy

» The frontend is written in ReactJS

» CERTitude supports many IOC standards: OpenIOC, STIX, CSV files, etc.
Step 1 – Create your hunting environment
Step 1 – Create your hunting environment
Step 1 – Create your hunting environment
Step 1 – Create your hunting environment
Step 2 – Design your collectors

Install and configure your centralizer(s)

» Depending on the network, you will have to create one or more centralizers, each one identified by its GUID and IP address

» A type of collector must be chosen depending on the target OS and the centralizer nodes configuration must be updated so that the collectors will know how to reach its API

Designing GPOs

» CERTitude offers an assist tool for creating Group Policy Objects (GPO) to launch the collectors on the target assets

» The GPO will have to be manually configured to ensure no write right for anyone but domain administrators on the executable or its configuration file

Technology sneak peak

» The centralizer nodes API is written in Python3 with Flask and SQLAlchemy

» The user frontend is written in ReactJS
Step 3 – Watch your collectors

What does your collector do?

» On Windows, collectors are launched at the asset’s initiative (since it is deployed by GPO) and executed as SYSTEM

» On Linux, collectors must be launched manually as root, or set to interface with a pre-existing agent (e.g. puppet)

» When launched, collectors declare themselves to their centralizer API and retrieve their GUID

» Then, they collect a few pieces of information about the OS (OS type, version, hostname, etc.) so that the centralizer node can quickly start tracking their status

» Finally, each collector module is launched in a separate thread (to allow some of them to potentially fail) and will report its encrypted results directly to the API

Technology sneak peak

» Collectors are written in Python 2 or 3 and packed using PyInstaller

» There are collectors for Windows Vista +, Windows XP and Unix assets

» Collectors are multithreaded and only write compressed and encrypted data to the disk
Step 3 – Watch your collectors

> collector.exe ..\config.json

[2019-10-29 16:03:00,454] collector INFO Collector has started
[2019-10-29 16:03:00,715] collector INFO Initialization successful, collector has been assigned GUID 682b81f7-378d-4f50-b6c1-2e7f1a1aeb3b
[2019-10-29 16:03:14,777] collector INFO Upload osinfo successful!
[2019-10-29 16:03:14,779] module.process INFO Running module
[2019-10-29 16:03:14,779] module.registry INFO Running module
[2019-10-29 16:03:14,990] module.process INFO Declaration successful, module has been assigned GUID f6d76b5b-9ebd-4bfa-bdf7-c807033b7f9b
[2019-10-29 16:03:14,993] module.registry INFO Declaration successful, module has been assigned GUID 62533247-a0bf-467b-a94d-00c7d5a9a40d
Step 3 – Watch your collectors

```bash
> collector.exe ..\config.json
[2019-10-29 16:03:00,454] collector INFO Collector has started
[2019-10-29 16:03:00,715] collector INFO Initialization successful, collector has been assigned GUID 682b81f7-6220-4f78-b903-09617b19f597
[2019-10-29 16:03:14,777] collector INFO Upload osinfo successful!
[2019-10-29 16:03:14,779] module.process INFO Running module
[2019-10-29 16:03:14,779] module.registry INFO Running module
[2019-10-29 16:03:14,990] module.process INFO Declaration successful, module has been assigned GUID f6d76b5b-9ebd-41d8-b707-97935358e951
[2019-10-29 16:03:14,993] module.registry INFO Declaration successful, module has been assigned GUID 625333247-a0bf-4fab-8a7c-2280b67e300b
```
Step 4 – Analyzing your results

The core of CERTitude

» Once the collection phase has started, results are uploaded frequently on the centralizer nodes through the API

» Dedicated analysis engines are running on the centralizing nodes to cross individual results with elementary indicators to produce matches

» Users able to log in on the centralizer node can monitor constantly the progress of the analyses, and reassess priority by indicator if needed

» At any time these users can export the centralizer node’s analyses results in order to upload them to the master node

» There, a visualizing interface will allow you to search for compromised hosts and crawl through the compromised data that is associated with them

Technology sneak peak

» Thanks to public-key cryptography, results can only be analyzed on the associated centralizer

» For the same reasons, analysis results cannot be forged by the attacker
What do we need?

Technology

» On Windows, CERTitude relies on Group Policy Objects (GPO) for its collector deployment

» Deploying CERTitude requires the ability to create GPOs on the target domain(s)
What do we need?

**Technology**

- On Windows, CERTitude relies on Group Policy Objects (GPO) for its collector deployment
- Deploying CERTitude requires the ability to create GPOs on the target domain(s)

**Communication**

- Collectors deployed on assets need to exchange data with centralizing nodes (tracking and data retrieval)
- Collectors communicate with the centralizing API using HTTPS – network rules must allow this traffic
What do we need?

**Technology**
- On Windows, CERTitude relies on **Group Policy Objects** (GPO) for its collector deployment
- Deploying CERTitude requires the ability to create **GPOs** on the target domain(s)

**Communication**
- Collectors deployed on assets need to **exchange data** with centralizing nodes (tracking and data retrieval)
- Collectors communicate with the centralizing **API** using **HTTPS** – network rules must allow this traffic

**Server & Storage**
- Intermediate centralizing nodes **retrieve** all **collected data** from their assets, in compressed and encrypted form
- These data must be stored on dedicated **hardened servers** with enough **storage space**
The many “plus sides” of CERTitude

Ensuring confidentiality all the way

» In DFIR cases you don’t want the attacker to know what indicators you are searching for ...
  » ... from the assets point of view
  » ... from the network point of view

» In other cases, your client’s data is highly sensitive and cannot be exfiltrated for off-site analyses (i.e. it must stay within a tight perimeter in the client IS)

» CERTitude answers both of these needs natively and painlessly:

IOC**s never go further than centralizer nodes**

Raw client data never go further than centralizer nodes.
The many “plus sides” of CERTitude

The “DAT” mystery

» Most of CERTitude tiers rely on DAT files: user.dat, node.dat, ioc.dat, and results.dat

» These DAT files embed JSON data that is exported straight from a tier’s database (e.g. the master’s) and imported into another tier’s database (e.g. the centralizer’s)

» Should not be a problem right?
The many “plus sides” of CERTitude

The “DAT” mystery

» Most of CERTitude tiers rely on DAT files: user.dat, node.dat, ioc.dat, and results.dat

» These DAT files embed JSON data that is exported straight from a tier’s database (e.g. the master’s) and imported into another tier’s database (e.g. the centralizer’s)

» Should not be a problem right? Of course this is not secure!

» That is why the master node and the centralizer nodes both have an asymmetric RSA key pair

» Key pairs are used to ...
  » ... decrypt sensitive data: CryptoCompress’d results from the hosts are decrypted that way
  » ... sign data to be imported another place (DAT files): user token, node configuration, analyses results, etc.
The many “plus sides” of CERTitude

The CryptoCompress library

» The collector and centralizer nodes use the Writer and Reader class of the CryptoCompress library, which has been developed exclusively for CERTitude.

» This library ensures that data collected with elevated privileges on the assets will not be read by lower-privileged accounts or intercepted during transfer.

» This library provides a open, read, write and close API that programs can use to write data on the disk, but:
  » Data is compressed first using a Zlib stream.
  » The output of stream is encrypted using state-of-the-art symmetric AEAD stream cipher ChaCha20-Poly1305.
  » Symmetric session keys are encrypted for each file using the centralizer node’s public RSA key.

Technology sneak peak

write()
Future roadmap

Short term (will be included in 3.0)

» Analysis engine v1.0

» Advanced results visualization (primary & secondary IOC)

» Always more collection modules (at least to match the previous versions’ capabilities)
Future roadmap

**Short term (will be included in 3.0)**

» Analysis engine v1.0

» Advanced results visualization (primary & secondary IOC)

» Always more collection modules (at least to match the previous versions’ capabilities)

**Medium & long term**

» Collected data explorer on centralizer nodes

» Yara support (dedicated collector)

» Campaign search & history

» Offline disk image & memory snapshot analysis
Future roadmap

**Short term (will be included in 3.0)**

» Analysis engine v1.0

» Advanced results visualization (primary & secondary IOC)

» Always more collection modules (at least to match the previous versions’ capabilities)

**Medium & long term**

» Collected data explorer on centralizer nodes

» Yara support (dedicated collector)

» Campaign search & history

» Offline disk image & memory snapshot analysis

+ Every idea YOU may have
Questions anybody?
Jean MARSAULT
CERT-W – Head of R&D

Vincent NGUYEN
Head of CERT-W

Jean.marsault@wavestone.com
+33 (0)7 60 67 93 34

Vincent.nguyen@wavestone.com
+33 (0)7 62 83 13 61

riskinsight-wavestone.com
@Risk_Insight

securityinsider-wavestone.com
@SecuInsider

wavestone.com
@wavestone_