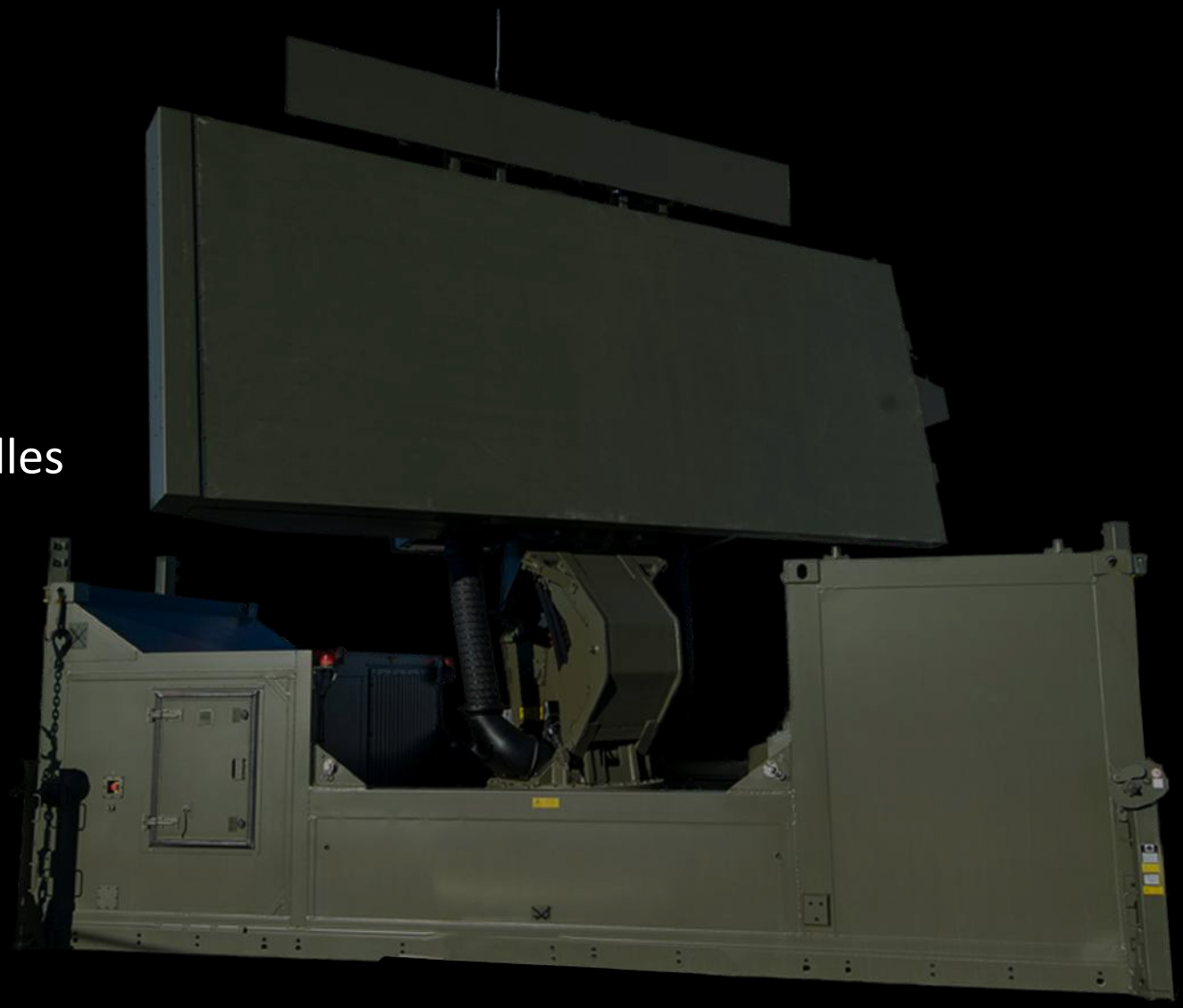
Projet E4 : Intégrité d’un RADAR Legacy

Chalubert Hugo , Dahmani Taha , Mehdi Youssef , Gao Angélique , Tsang Bryan

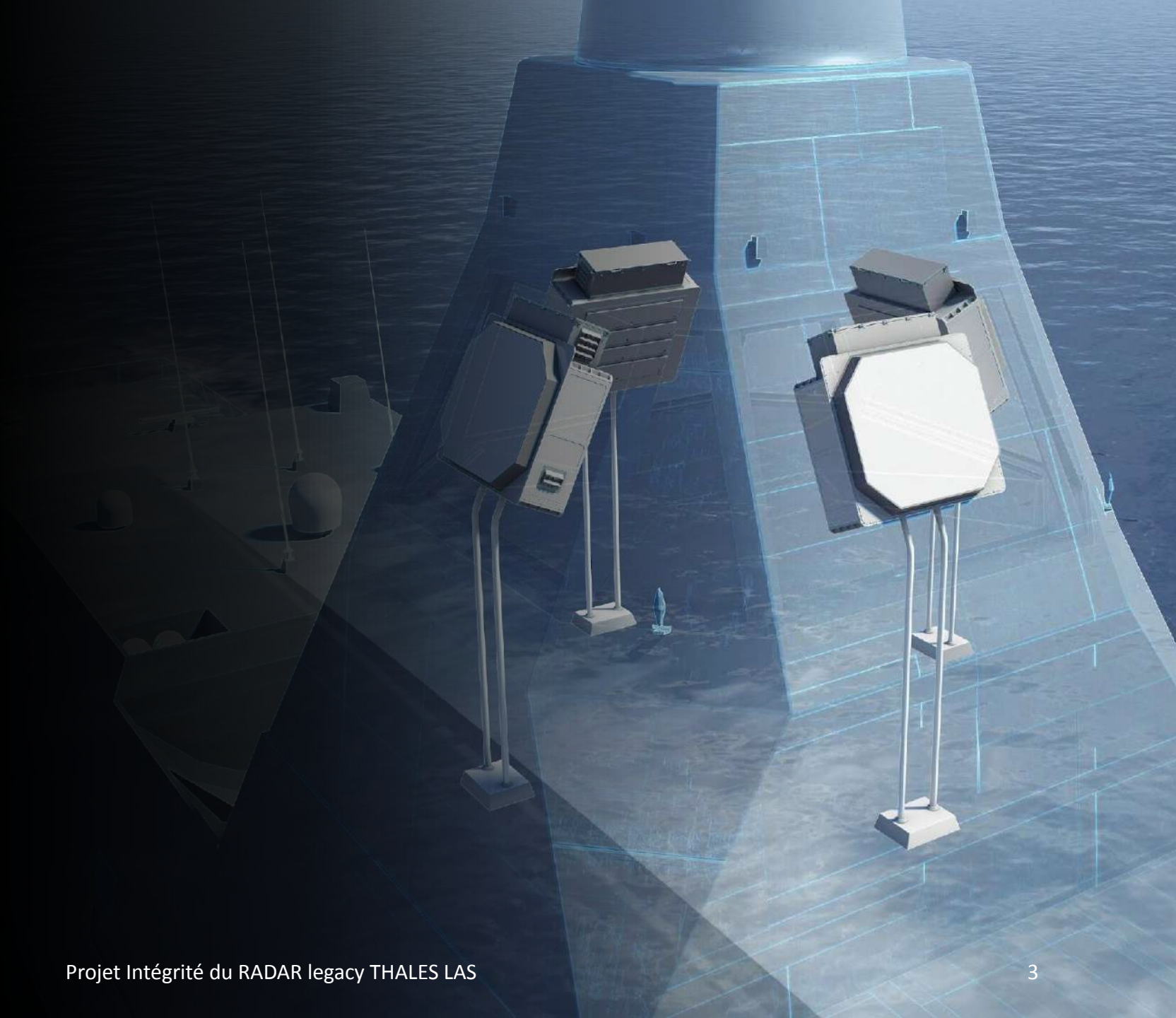
Sommaire

- Configuration
- Cybersécurité détection & exploitation des failles
- Caldera
- Surveillance par l’IA
- Conclusion

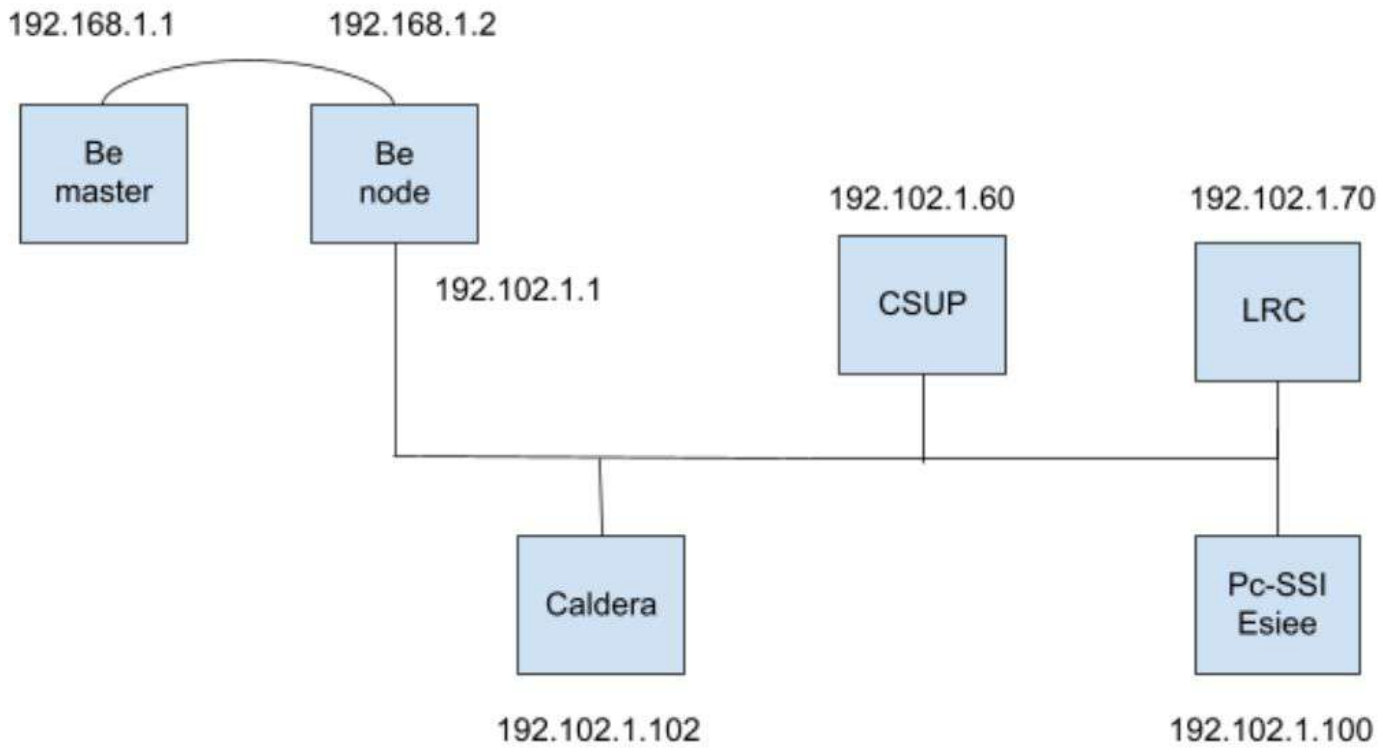


Configuration des VMs

- Architecture réseau
- Configuration des VM sur virtualbox
- Complication sur la configuration de la LRC
- Complication sur la Caldera red-team



Architecture réseau



Complication sur la LRC

- IRQ
- I/O APIC
- Correction
- console=tty0 console=ttyS0,115200 divider=10 noapic
- pci=routeirq

```
root (hd0,0)
kernel /vmlinuz-2.6.18-194.el5PAE ro root=/dev/VolGroup00/LogVol00
initrd /initrd-2.6.18-194.el5PAE.img
```

```
ACPI: PCI Interrupt Link [LNKC] enabled at IRQ 9
ACPI: PCI Interrupt 0000:00:03.0[A] -> Link [LNKC] -> GSI 9 (level, low) -> IRQ
9
ACPI: PCI Interrupt Link [LNKD] enabled at IRQ 11
ACPI: PCI Interrupt 0000:00:04.0[A] -> Link [LNKD] -> GSI 11 (level, low) -> IRQ
 11
ACPI: PCI Interrupt Link [LNKA] enabled at IRQ 11
ACPI: PCI Interrupt 0000:00:05.0[A] -> Link [LNKA] -> GSI 11 (level, low) -> IRQ
 11
ACPI: PCI Interrupt 0000:00:07.0[A] -> Link [LNKC] -> GSI 9 (level, low) -> IRQ
9
```

```
Total of 1 processors activated (5606.40 BogoMIPS).
ENABLING IO-APIC IRQs
..TIMER: vector=0x31 apic1=0 pin1=2 apic2=-1 pin2=-1
..MP-BIOS bug: 8254 timer not connected to IO-APIC
...trying to set up timer (IRQ0) through the 8259A ...  failed.
...trying to set up timer as Virtual Wire IRQ... failed.
...trying to set up timer as ExtINT IRQ... failed :(.
Kernel panic - not syncing: IO-APIC + timer doesn't work!  Boot with apic=debug
and send a report.  Then try booting with the 'noapic' option
```

Complication sur la VM Caldera

- Serveur.py
- Non détection des VM
- Des habilités non-détectées
- Kernel panic
- Problème de virtualisation

```
    ide1: BM-DMA at 0xc008-0xc00f, BIOS settings: hdc:pio, hdd:pio
ne2k-pci.c:v1.03 9/22/2003 D. Becker/P. Gortmaker
  http://www.scyld.com/network/ne2k-pci.html
hda: QEMU HARDDISK, ATA DISK drive
ide0 at 0x1f0-0x1f7,0x3f6 on irq 14
hdc: QEMU CD-ROM, ATAPI CD/DVD-ROM drive
ide1 at 0x170-0x177,0x376 on irq 15
ACPI: PCI Interrupt Link [LNKC] enabled at IRQ 10
ACPI: PCI Interrupt 0000:00:03.0[A] -> Link [LNKC] -> GSI 10 (level, low) -> IRQ
 10
eth0: RealTek RTL-8029 found at 0xc100, IRQ 10, 52:54:00:12:34:56.
hda: max request size: 512KiB
hda: 180224 sectors (92 MB) w/256KiB Cache, CHS=178/255/63, (U)DMA
hda: set_multmode: status=0x41 { DriveReady Error }
hda: set_multmode: error=0x04 { DriveStatusError }
ide: failed opcode was: 0xef
hda: cache flushes supported
 hda: hda1
hdc: ATAPI 4X CD-ROM drive, 512kB Cache, (U)DMA
Uniform CD-ROM driver Revision: 3.20
Done.
Begin: Mounting root file system... ...
/init: /init: 151: Syntax error:  0xforce=panic
Kernel panic - not syncing: Attempted to kill init!
 _
```

Cybersécurité

- Détection et exploitation des failles de sécurité (sur la machine de supervision)
  - Mise en place
  - Scan sans identifiants ssh
  - Scan avec identifiants ssh
  - Pentest manuel avec la version du kernel

Cybersécurité - Mise en place

```
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
    link/ether 08:00:27:37:fc:bd brd ff:ff:ff:ff:ff:ff
    inet 192.102.1.60/24 brd 192.102.1.255 scope global enp0s3
       valid_lft forever preferred_lft forever
    inet6 fe80::a00:27ff:fe37:fcbd/64 scope link
       valid_lft forever preferred_lft forever
```

Nessus®
vulnerability scanner

Greenbone

Cybersécurité - Scan w/o credentials

Atlas CSup w/o credentials	192.102.1.60	1	All IANA assigned TCP

Hosts

Included	192.102.1.60
Maximum Number of Hosts	1
Allow simultaneous scanning via multiple IPs	Yes
Reverse Lookup Only	No
Reverse Lookup Unify	No
Alive Test	Scan Config Default
Port List	All IANA assigned TCP

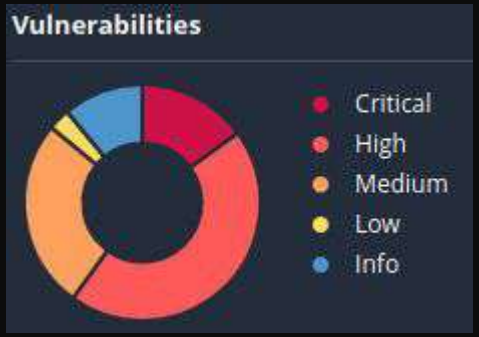
Edit Task Full scan CSup w/o credentials

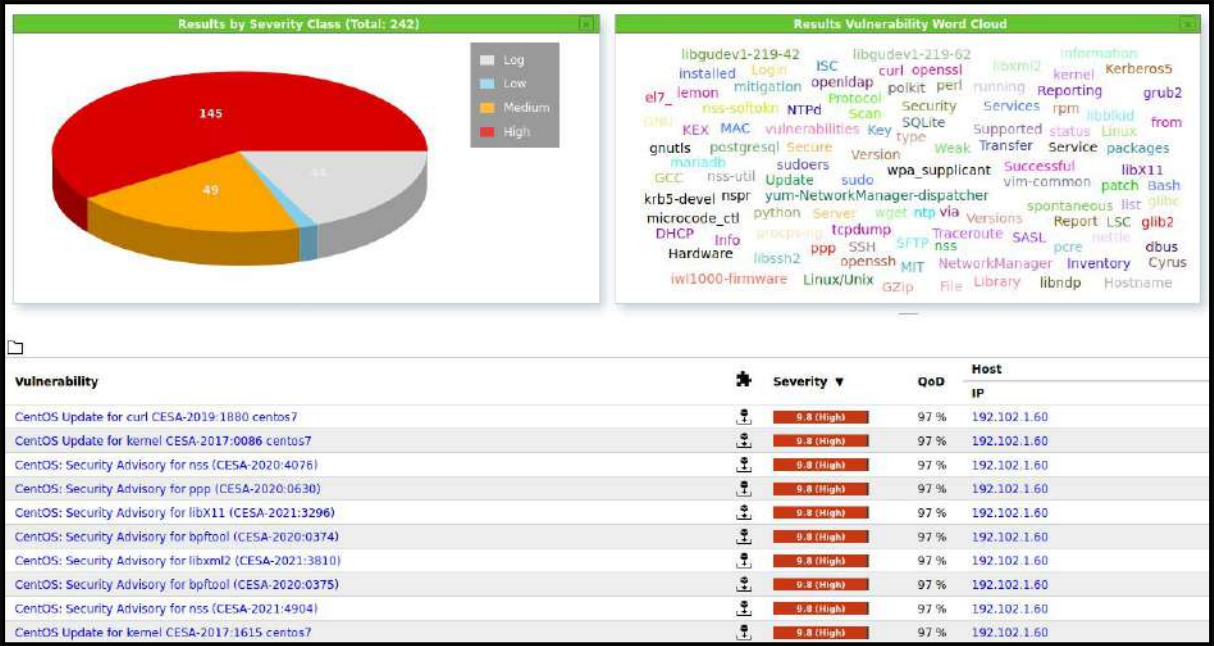
Name	Full scan CSup w/o credentials
Comment	
Scan Targets	Atlas CSup w/o credentials
Alerts	
Schedule	-- Once
Add results to Assets	Yes No
Apply Overrides	Yes No
Min QoD	70 %
Auto Delete Reports	Do not automatically delete reports Automatically delete oldest reports but always keep newest 5 reports
Scanner	OpenVAS Default
Scan Config	Full and fast
Network Source Interface	

Cybersécurité - Scan w/o credentials

Vulnerability	Severity ▼	QoD	Host IP
Weak Key Exchange (KEX) Algorithm(s) Supported (SSH)	5.3 (Medium)	80 %	192.102.1.60
Weak Encryption Algorithm(s) Supported (SSH)	4.3 (Medium)	95 %	192.102.1.60
Weak MAC Algorithm(s) Supported (SSH)	2.6 (Low)	95 %	192.102.1.60
SSH Protocol Algorithms Supported	0.0 (Log)	80 %	192.102.1.60
SSH Protocol Versions Supported	0.0 (Log)	95 %	192.102.1.60
OS Detection Consolidation and Reporting	0.0 (Log)	80 %	192.102.1.60
Traceroute	0.0 (Log)	80 %	192.102.1.60
CPE Inventory	0.0 (Log)	80 %	192.102.1.60
Services	0.0 (Log)	80 %	192.102.1.60
Hostname Determination Reporting	0.0 (Log)	80 %	192.102.1.60

Cybersécurité - Scan w/ credentials





Cybersécurité - Pentest

```
Linux ATLAS-CSUP.centos.7_2 3.10.0-327.el7.x86_64 #1 SMP Thu Nov 19 22:10:57 UTC 2015 x86_64 x86_64
x86_64 GNU/Linux
```

```
┌──(kali㉿kali)-[~]
└─$ searchsploit centos 3.10
---------------------------------------------------------------------------------------------------
 Exploit Title
---------------------------------------------------------------------------------------------------
Linux Kernel 2.6.x / 3.10.x / 4.14.x (RedHat / Debian / CentOS) (x64) - 'Mutagen Astronomy' Local Privilege Escalation
Linux Kernel 3.10.0 (CentOS / RHEL 7.1) - 'aiptek' Nullpointer Dereference
Linux Kernel 3.10.0 (CentOS / RHEL 7.1) - 'cdc_acm' Nullpointer Dereference
Linux Kernel 3.10.0 (CentOS / RHEL 7.1) - 'cypress_m8' Nullpointer Dereference
Linux Kernel 3.10.0 (CentOS / RHEL 7.1) - 'digi_acceleport' Nullpointer Dereference
Linux Kernel 3.10.0 (CentOS / RHEL 7.1) - 'mct_u232' Nullpointer Dereference
Linux Kernel 3.10.0 (CentOS / RHEL 7.1) - 'Wacom' Multiple Nullpointer Dereferences
Linux Kernel 3.10.0 (CentOS / RHEL 7.1) - visor 'treo_attach' Nullpointer Dereference
Linux Kernel 3.10.0 (CentOS / RHEL 7.1) - visor clie_5_attach Nullpointer Dereference
Linux Kernel 3.10.0 (CentOS 7) - Denial of Service
Linux Kernel 3.10.0-229.x (CentOS / RHEL 7.1) - 'iowarrior' Driver Crash (PoC)
Linux Kernel 3.10.0-229.x (CentOS / RHEL 7.1) - 'snd-usb-audio' Crash (PoC)
Linux Kernel 3.10.0-514.21.2.el7.x86_64 / 3.10.0-514.26.1.el7.x86_64 (CentOS 7) - SUID Position Independent Executable 'PIE' Local Privilege Escalation
```

Cybersécurité

Exploit ‘Mutagen Astronomy’

```
[test_exploit@ATLAS-CSUP exploit]$ gcc exploit_45516.c -o exploit_45516
[test_exploit@ATLAS-CSUP exploit]$ gcc poc-suidbin.c -o poc-suidbin
[test_exploit@ATLAS-CSUP exploit]$ ./exploit_45516
died in main: 233
[test_exploit@ATLAS-CSUP exploit]$ whoami
test_exploit
[test_exploit@ATLAS-CSUP exploit]$
```

Exploit ‘PIE’

```
[test_exploit@ATLAS-CSUP exploit]$ gcc -fpic -shared -nostartfiles -Os -s -o rootshell rootshell.c
[test_exploit@ATLAS-CSUP exploit]$ xxd -i rootshell > rootshell.h
[test_exploit@ATLAS-CSUP exploit]$ gcc 42887.c -o exploit_42887
[test_exploit@ATLAS-CSUP exploit]$ ./exploit_42887
Usage: ./exploit_42887 binary
died in main: 204
```

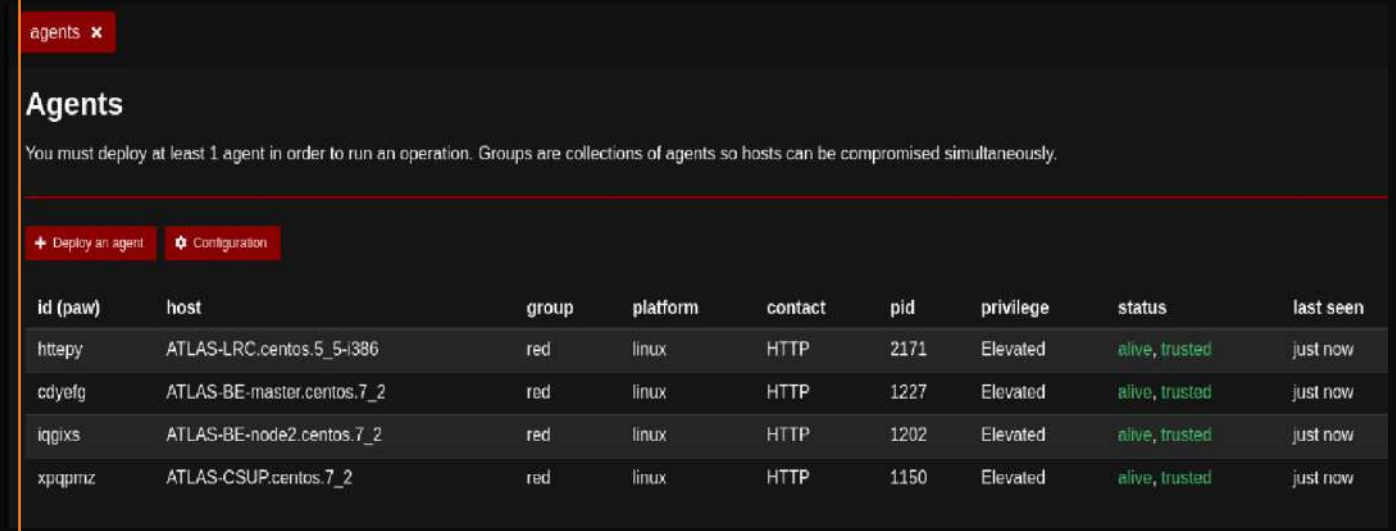
Exploit Denial of Service

```
┌──(kali㉿kali)-[~]
└─$ sudo ./exploit_dos.out -T3 -h 192.102.1.60 -p [514,514]
```

Caldera Red Team

- Déployer des agents
- Détecter les agents en service depuis la LRC
- Créer un planner
- Créer une habilité
- Lancer des opérations complexes

Déployer des agents

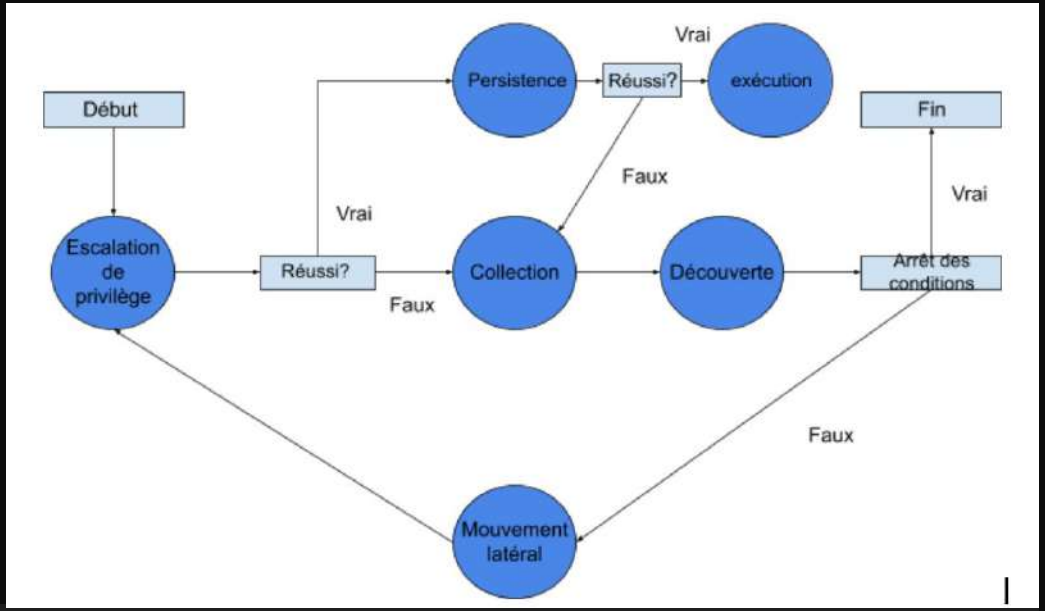


id (paw)	host	group	platform	contact	pid	privilege	status	last seen
httepy	ATLAS-LRC.centos.5_5-i386	red	linux	HTTP	2171	Elevated	alive, trusted	just now
cdyefg	ATLAS-BE-master.centos.7_2	red	linux	HTTP	1227	Elevated	alive, trusted	just now
iqgixs	ATLAS-BE-node2.centos.7_2	red	linux	HTTP	1202	Elevated	alive, trusted	just now
xpqpmz	ATLAS-CSUP.centos.7_2	red	linux	HTTP	1150	Elevated	alive, trusted	just now

Affiche l’ensemble des agents connectés s’ils sont prêt à recevoir des opérations

Création du planner

- En python & Bash
- Une FSM (final state machine)
- Ajout de persistance




```python
async def persistence(self):
    self.log.info('starting to get persistence')
    await self.planning_svc.exhaust_bucket(self, 'persistence', self.operation)
    successful = await self.has_been_modified()
    if not successful:
        self.next_bucket = await self.planning_svc.default_next_bucket('collection', self.state_machine)
    elif successful:
        self.next_bucket = await self.planning_svc.default_next_bucket('execution', self.state_machine)
```

```python
async def collection(self):
    self.log.info('starting to get collection')
    await self.planning_svc.exhaust_bucket(self, 'collection', self.operation)
    self.next_bucket = 'discovery'

async def execution(self):
    self.log.into('trying to make some execution')
    await self.planning_svc.exhaust_bucket(self, 'execution', self.operation)
    self.next_bucket = await self.planning_svc.default_next_bucket('collection', self.state_machine)

async def discovery(self):
    self.log.info('starting discovery state')
    await self.planning_svc.exhaust_bucket(self, 'discovery', self.operation)
    lateral_movement_unlocked = bool(len(await self.planning_svc.get_links(self.operation, buckets=['la
    if lateral_movement_unlocked:
        self.next_bucket='lateral_movement'
    else:
        self.next_bucket= None

async def lateral_movement(self):
    self.log.info('starting to make lateral movement')
    await self.planning_svc.exhaust_bucket(self, 'lateral-movement',self.operation)
    self.next_bucket= 'privilege_escalation'
```

Création d’une habilité

- En python & Bash
- Il s’agit d’un fichier yml
- Il contient les commandes personnalisées

create-planner.sh | ac2d23b6-e3a2-4212-b064-4d4a05d00a69

```yml
---

- id: ac2d23b6-e3a2-4212-b064-4d4a05d00a69
  name: red
  description: find every commands with SUID place to 1
  tactic: discovery
  technique:
    attack_id: T9999
    name: red
  platforms:
    linux:
      sh:
        command: |
          find / -perm -u=s -type f 2>/dev/null
```

+ Add Ability | + Add Adversary | Objective: **default** | Change | Save Profile | Delete Profile

	Ordering	Name	Tactic	Technique	Executors	Requires	Unlocks	Payload	Cleanup
≡	1	Create a new user in Linux with `root` UID and GID.	persistence	Create Account: Local Account					
≡	2	Find files	collection	Data from Local System					
≡	3	Create and Execute Bash Shell Script	execution	Command and Scripting Interpreter: Bash					
≡	4	Start 54ndc47	lateral-movement	Remote Services: SSH					
≡	5	System Network Configuration Discovery	discovery	System Network Configuration Discovery					
≡	6	Examine password complexity policy - CentOS/RHEL 7.x	discovery	Password Policy Discovery					
≡	7	Examine password complexity policy - CentOS/RHEL 6.x	discovery	Password Policy Discovery					
≡	8	USB Connected Device Discovery	discovery	Peripheral Device Discovery					
≡	9	Weak executable files	privilege-escalation	Hijack Execution Flow: Services File Permissions Weakness					

Search for an ability...

Tactic: discovery

Technique: T9999 | red

Ability: red

Lancer une opération

- Exécute pas à pas les états du planner
- Les états viennent récupérer 1 à 1 les habilités
- Affiche les résultats
- Possibilité de l’exécuter grâce à curl

Caldera Blue Team

- Blue team
- Abilities Caldera Blue team
- Lancement de l’opération

Blue Team

- Préparation
- Identification
- Endiguement
- Eradication
- Récupération
- Leçons apprises

Abilities Blue team

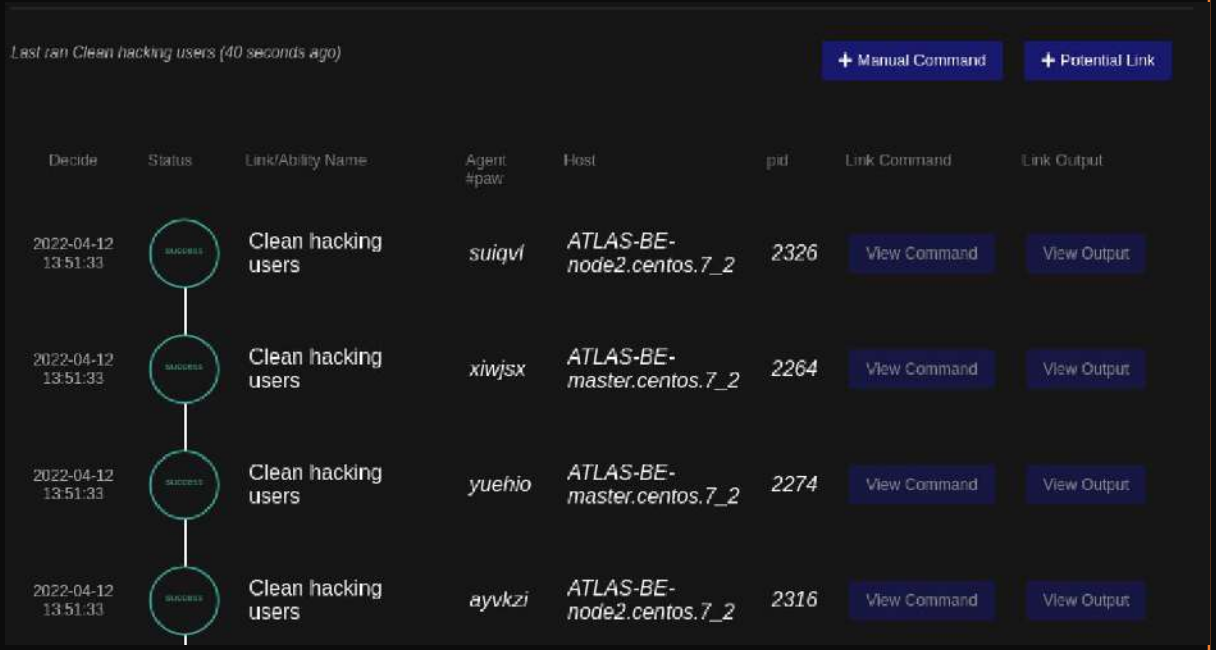
RADAR Blue-Team

A basic Radar Blue TEAM profile

+ Add Ability | + Add Adversary | Objective: **default** | Change | Save Profile | Delete Profile

	Ordering	Name	Tactic	Technique	Executors	Requires	Unlocks	Payload	Cleanup	
☰	1	Clean hacking users	clean	CLEAN Account: Local Account						×
☰	2	Hash Sensitive Files	setup	x						×
☰	3	Hash Sensitive Directories	setup	x						×
☰	4	Backup Sensitive Files	setup	x						×
☰	5	Backup Sensitive Directories	setup	x						×
☰	6	Modified Sensitive Files	detection	x						×
☰	7	Modified Sensitive Directory	detection	x						×
☰	8	Restore File Backup	response	x						×

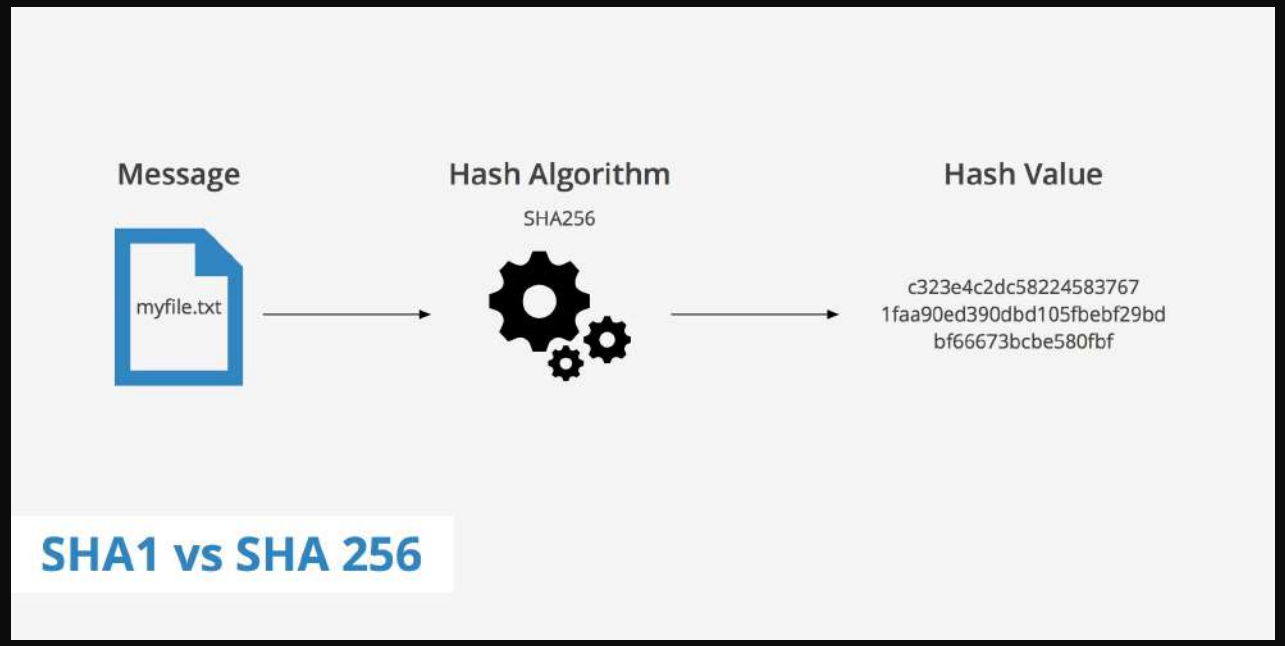
Lancement de l'opération




```
2022-04-12 11:53:50 - INFO  (blue.py:18 __init__) blue planner INIT !
2022-04-12 11:53:50 - INFO  (blue.py:25 execute) blue planner EXECUTE !
2022-04-12 11:53:50 - INFO  (blue.py:44 reset) blue planner RESET !
2022-04-12 11:54:40 - INFO  (blue.py:53 register) blue planner REGISTER !
2022-04-12 11:55:50 - INFO  (blue.py:61 detect) blue planner DETECT [1] !
2022-04-12 11:56:45 - INFO  (blue.py:71 detect) blue planner DETECT no has_been_modified!
2022-04-12 11:56:45 - INFO  (blue.py:61 detect) blue planner DETECT [2] !
2022-04-12 11:57:40 - INFO  (blue.py:71 detect) blue planner DETECT no has_been_modified!
2022-04-12 11:57:40 - INFO  (blue.py:61 detect) blue planner DETECT [3] !
2022-04-12 11:58:25 - INFO  (blue.py:71 detect) blue planner DETECT no has_been_modified!
2022-04-12 11:58:25 - INFO  (blue.py:61 detect) blue planner DETECT [4] !
2022-04-12 11:59:31 - INFO  (blue.py:71 detect) blue planner DETECT no has_been_modified!
2022-04-12 11:59:31 - INFO  (blue.py:61 detect) blue planner DETECT [5] !
2022-04-12 12:00:51 - INFO  (blue.py:31 has_been_modified) has_been_modified detected on /etc/passwd
2022-04-12 12:00:51 - INFO  (blue.py:67 detect) blue planner DETECT has_been_modified!
2022-04-12 12:00:51 - INFO  (blue.py:78 recovery) blue planner RECOVERY !
2022-04-12 12:00:51 - INFO  (blue.py:61 detect) blue planner DETECT [1] !
```

HASH Python

- Utilisation de hashlib
- Sha-256
- Permet l'intégrité de l'ensemble des fichiers
- Basé sur les premiers anti-virus
- En python



Surveillance du radar par l’Intelligence Artificielle

Création des sondes et envoi des données

- Choix des données à envoyer
- Acquisition des données
- Affranchissement de mot de passe afin d'automatiser l'envoi
- Installation sur la machine RIF_LRC

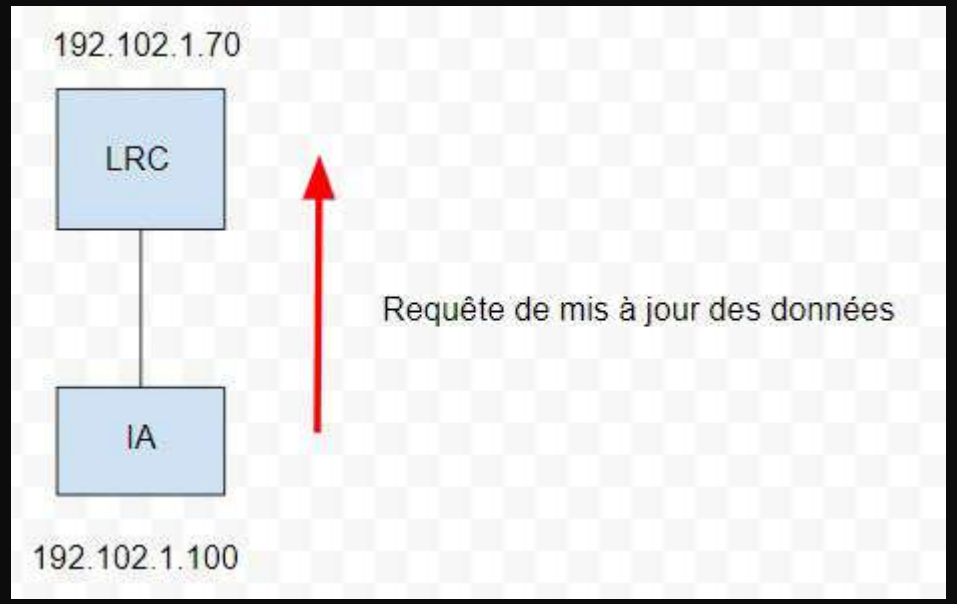
```bash
#!/bin/bash

ADDR=$1

ssh-keygen -t rsa -b 4096
ssh $ADDR


cat .ssh/id_rsa.pub | ssh $ADDR 'cat >> .ssh/authorized_keys'
ssh $ADDR
```

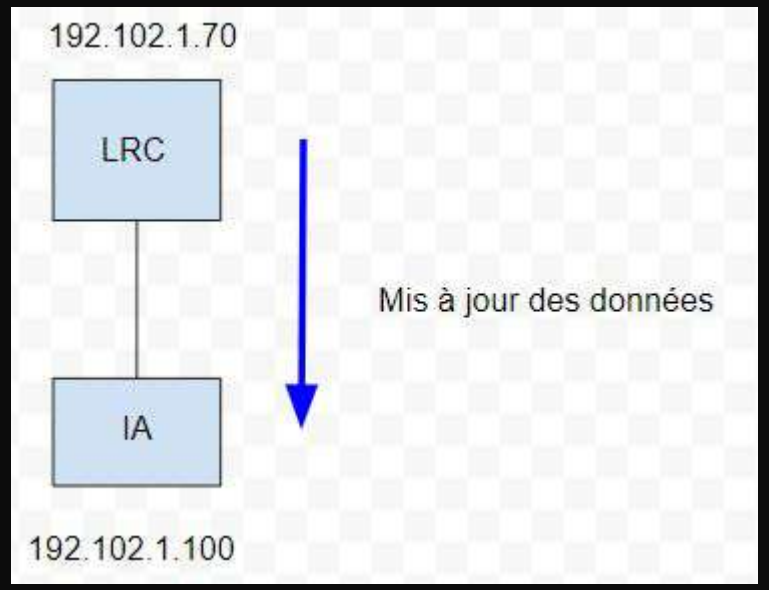
```bash
#Gather process datas via ps command and store them into a csv file
rm *.csv

ps ax o user --sort pid | tee user.csv
ps ax o %cpu --sort pid | tee cpu.csv
ps ax o %mem --sort pid | tee mem.csv
ps ax o pid --sort pid | tee pid.csv
ps ax o cmd --sort pid | tee cmd.csv
```

```bash
#send the previous data to the AI server
scp -p *.csv root@192.102.1.100:/root/Documents/DataCollection
```

Routine de communication

- Création d'une communication entre un serveur (IA) et un client (radar)
- Rassemblement des données sur le serveur
- Routine de requête et réponse entre les machines





root@ATLAS-LRC:~

File Edit View Search Terminal Help

```
[root@ATLAS-LRC ~]# ./client.o
I am client
process done
```

root@centos-7_8_2003-x86_64:~/Documents/DataCollection/TCP_IP_C

File Edit View Search Terminal Help

```
[root@centos-7_8_2003-x86_64 TCP_IP_C]# ./servertest.o
I am server
01
```

DataCollection data

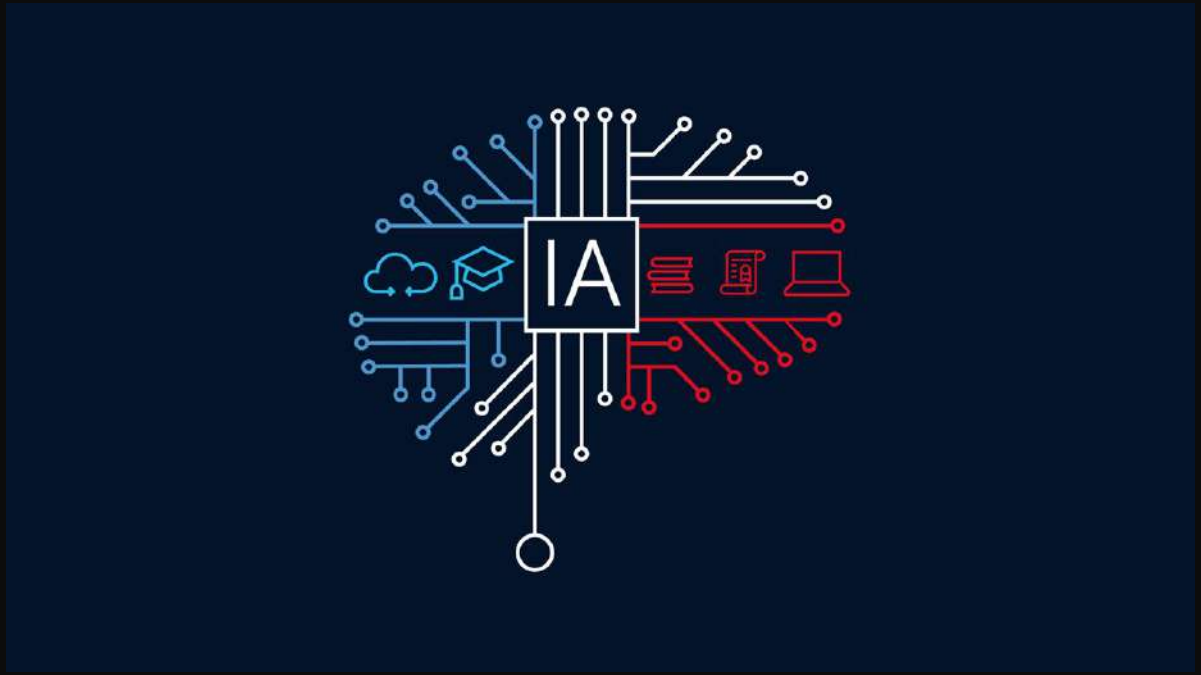
Recent
Home
Documents
Downloads
Music
Pictures
Videos
Trash
sf_SharedMemory
Other Locations

cmd.csv
cpu.csv
mem.csv
memUsage.csv
pid.csv
user.csv

CTRL DROITE

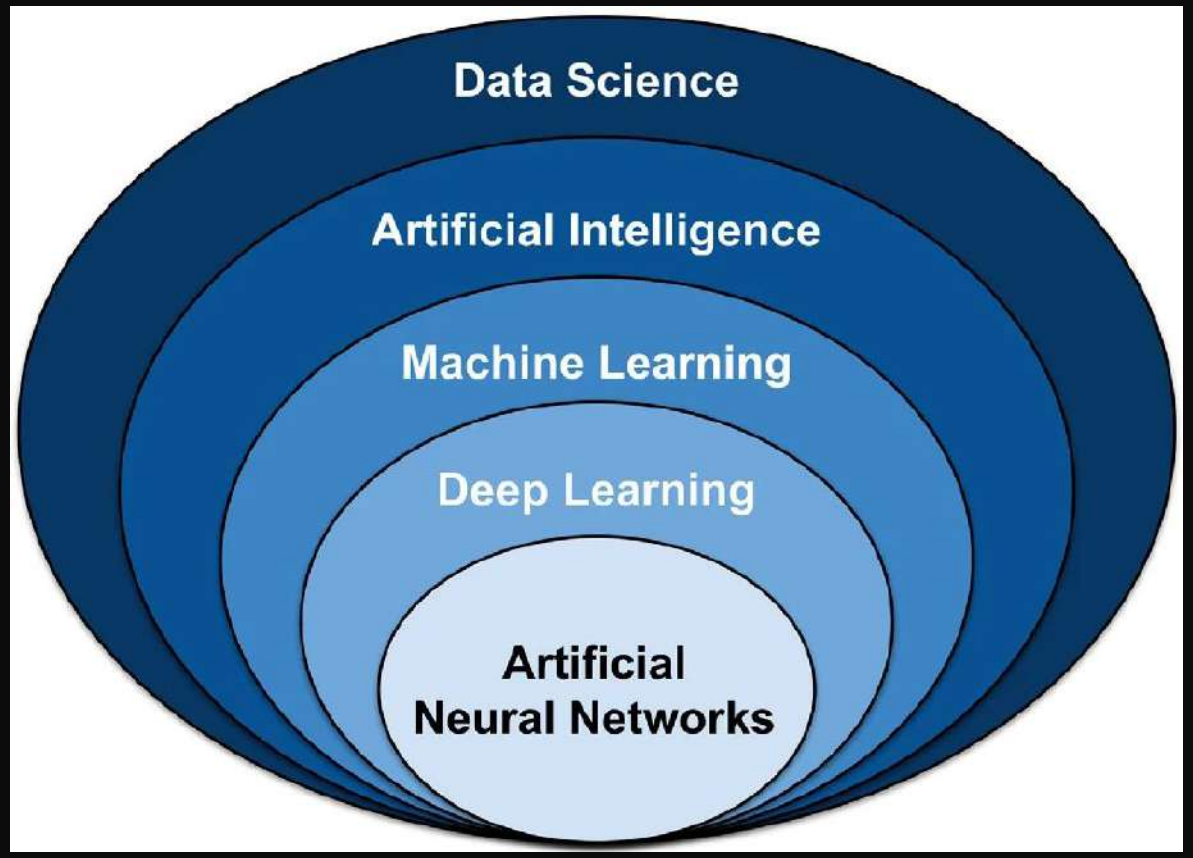
• Why AI?

- Identify Attacks patterns
- More accurate than people
- Faster
- Frees cyber engineers to focus on other complex tasks



Create models

- Normalize the data and delete unnecessary rows or columns.
- Split our dataset to Training and testing sets
- Reshape our data to be ready for the training
- Create the model
- Train the model
- Test the model



• Used Models

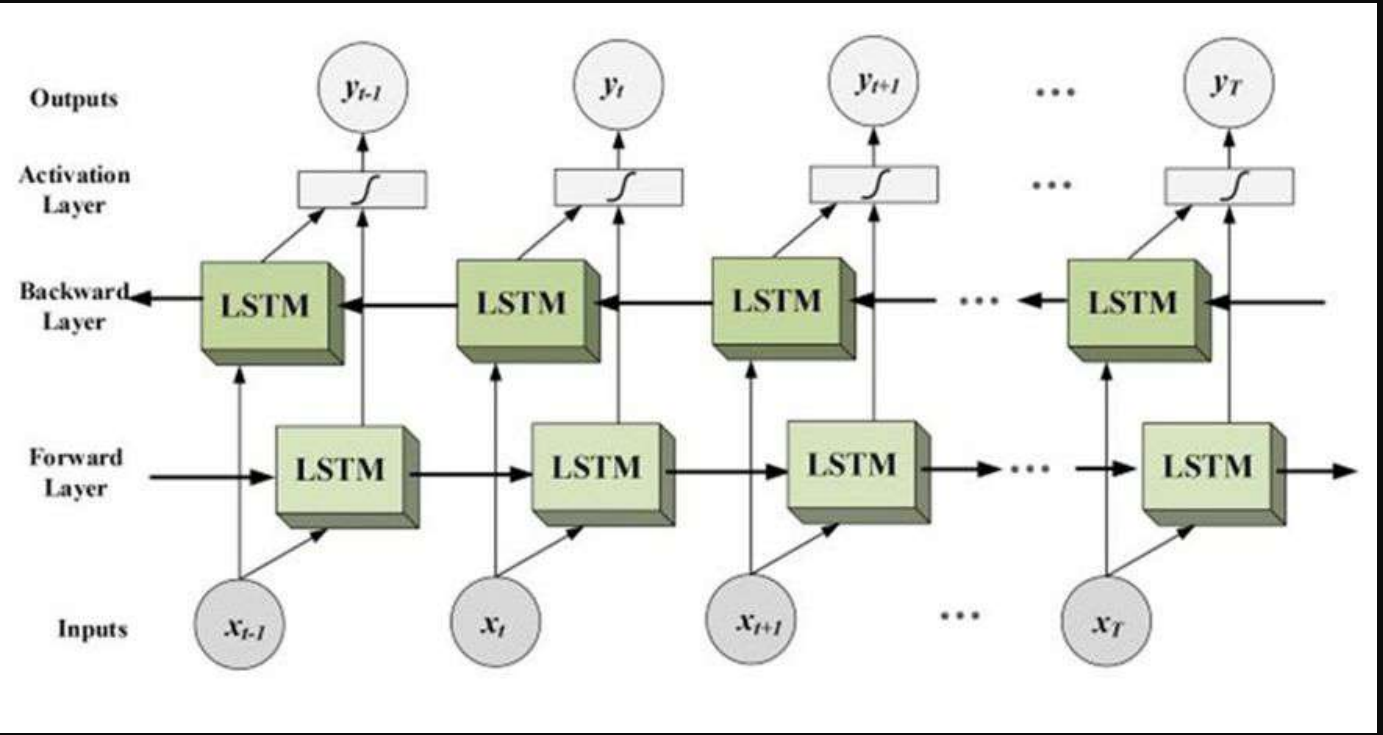
- Bi-LSTM (Deep Learning)
- Random Forest Classifier (Machine learning)

• Attacks to be detected:

- DDOs Attack
- Mirai Attack
- Scan Attack
- Spoofing

• Bi-LSTM

- Bidirectional long-short term memory(bi-lstm)



bidirectional_1(lstm_1): Bidirectional(LSTM)	input:	multiple
	output:	multiple

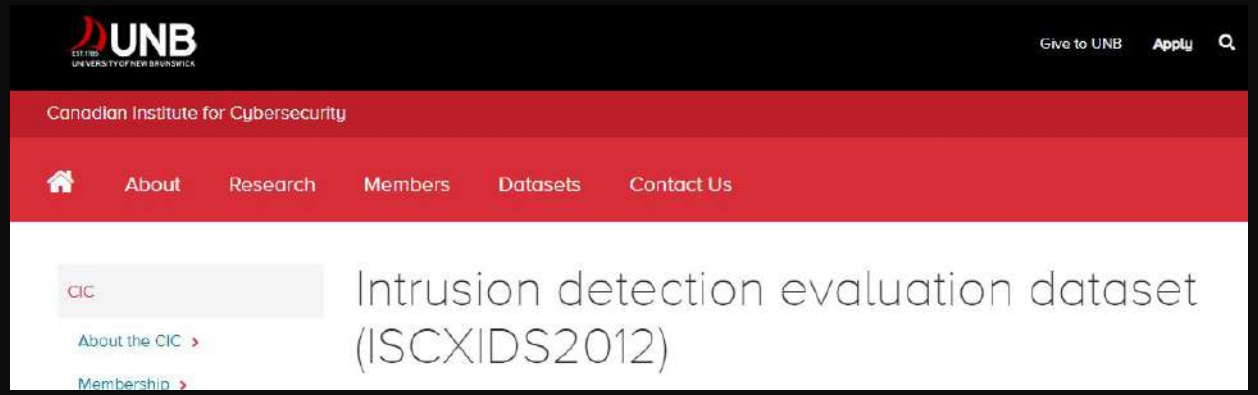
dense_1: Dense	input:	multiple
	output:	multiple

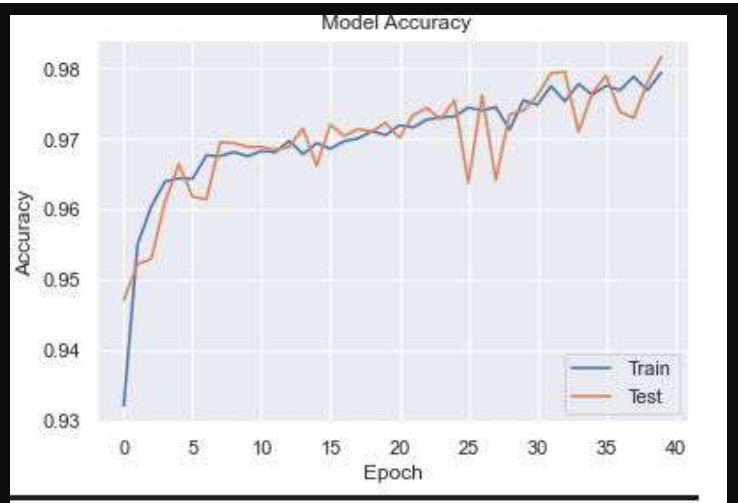
dense_2: Dense	input:	multiple
	output:	multiple

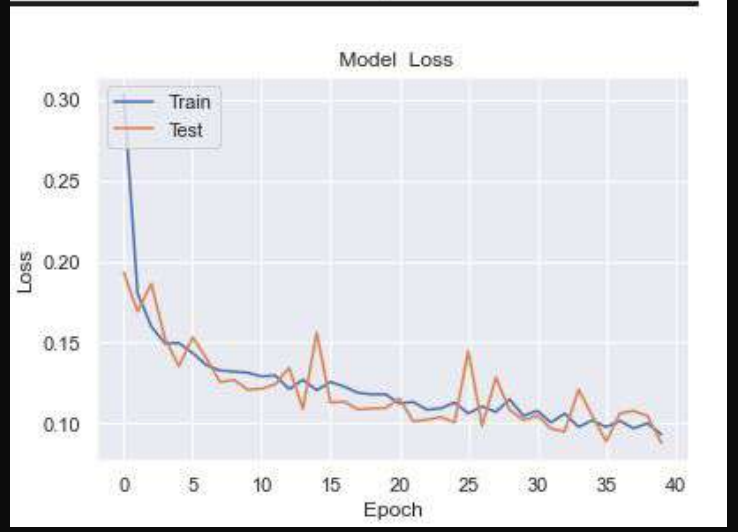
• BI-LSTM:

- Bi-LSTM Was used to detect the DDOs attacks on our system with accuracy of 98.18%

```
scores = model.evaluate(X_test, Y_test, verbose=0)
print("%s: %.2f%%" % (model.metrics_names[1], scores[1]*100))
✓ 2.3s

accuracy: 98.18%
```

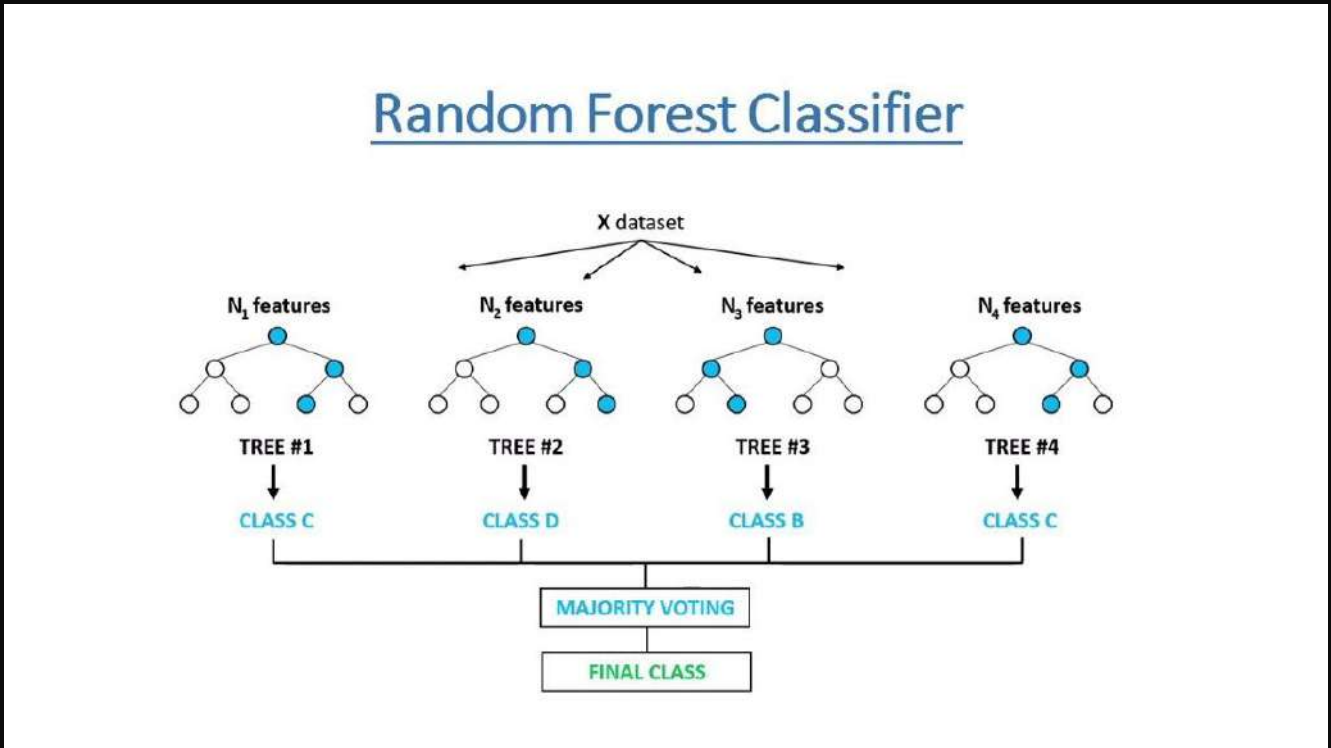






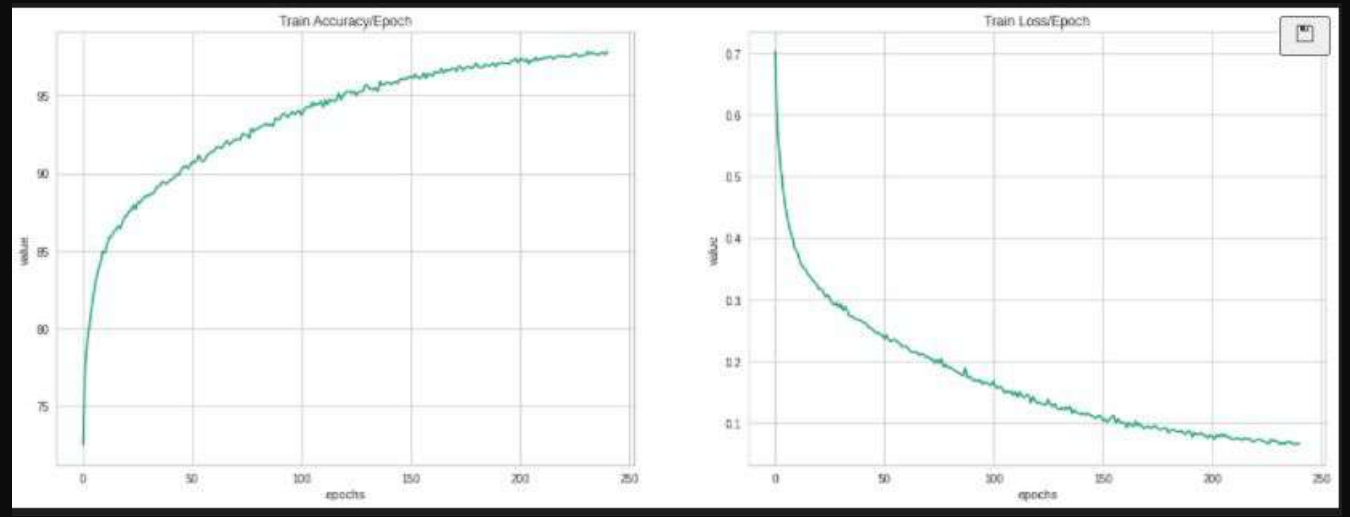
Random Forest Classifier

- Random forest is a supervised learning algorithm that builds "forests", it's an ensemble of decision trees, usually trained with the “bagging” method.



Random Forest Classifier

- Random Forest Classifier was used to detect Mirai, Scan and Spoofing attacks.



	precision	recall	f1-score	support
DoS	0.9954	0.9909	0.9932	220
Mirai	0.9607	0.9645	0.9626	1266
Normal	0.9577	0.9510	0.9544	143
Scan	0.9145	0.9386	0.9264	228
Spoofing	0.7350	0.6825	0.7078	126
accuracy			0.9455	1983
macro avg	0.9127	0.9055	0.9089	1983
weighted avg	0.9447	0.9455	0.9450	1983

Merci de votre écoute
Avez-vous des questions ?