SECURITY DOCUMENTATION

Team ID: 0032

System Overview

The system consists of 6 interconnected computers facilitating internal and external communication:

1) PLC (Programmable Logic Controller)
2) ICS CnC (Industrial Control System Command and Control)
3) Web Server
4) Public Database
5) Task Box
6) AD/DNS (Active Directory/Domain Name System)

Together these devices serve as the backbone of Energia Ventosa's operation tracking energy production, monitoring turbine status, and facilitating energy distribution in the company's Area of Responsibility (AOR).

The ICS CnC and PLC computers enable seamless communication with the wind turbines. The PLC system communicates directly with the energy infrastructure, collecting and transmitting data to the ICS CnC. This data is then accessed by the Task Box, which presents key metrics on a dashboard for operational oversight.

The Public Database and Web Server systems work in tandem to host a website that serves as the organization's official landing page, which doubles as a corporate web portal, providing employee login functionality and external visibility.

The AD/DNS machine provides domain services for the Ventosa and Power domains enabling single sign-on for all connected machines, enforcing security policies, and facilitating user-friendly navigation via the Domain Name Service (DNS).

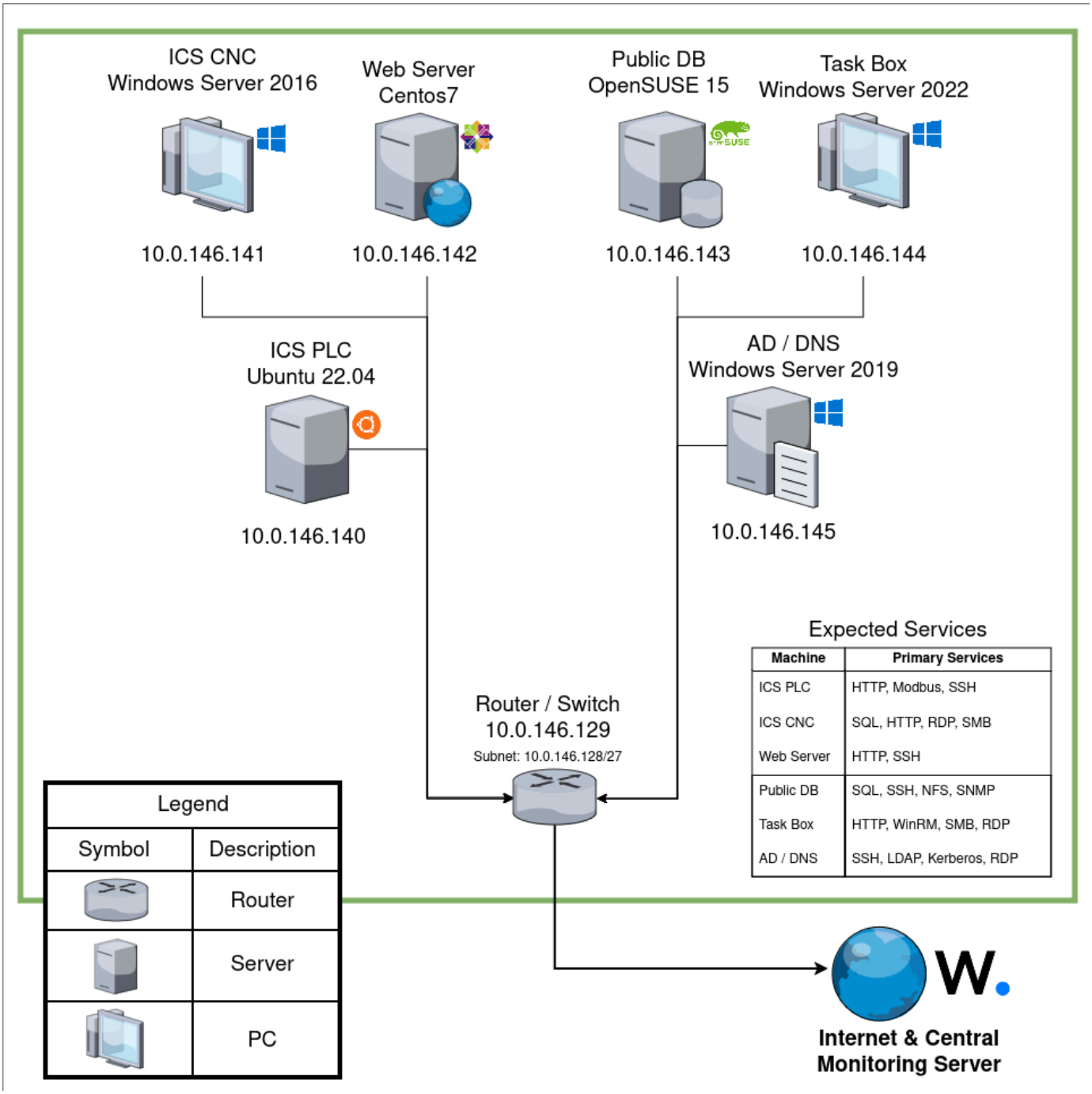
Asset Inventory

Host	OS	IP Address	Port	Service
Assume Breach				
PLC	Ubuntu 22.04	10.0.146.140	22/tcp	SSH
PLC	Ubuntu 22.04	10.0.146.140	80/tcp	HTTP
PLC	Ubuntu 22.04	10.0.146.140	502/tcp	Modbus
PLC	Ubuntu 22.04	10.0.146.140	8080/tcp	HTTP
PLC	Ubuntu 22.04	10.0.146.140	44818/tcp	EtherNet-IP 2
PLC	Ubuntu 22.04	10.0.146.140	20000/tcp	DNP3
CNC	Windows Server 2016	10.0.146.141	22/tcp	SSH
CNC	Windows Server 2016	10.0.146.141	53/tcp	DNS
CNC	Windows Server 2016	10.0.146.141	53/udp	DNS
CNC	Windows Server 2016	10.0.146.141	80/tcp	HTTP
CNC	Windows Server 2016	10.0.146.141	88/tcp	Kerberos
CNC	Windows Server 2016	10.0.146.141	135/tcp	MS-RPC
CNC	Windows Server 2016	10.0.146.141	139/tcp	NetBIOS-SSN
CNC	Windows Server 2016	10.0.146.141	389/tcp	LDAP
CNC	Windows Server 2016	10.0.146.141	443/tcp	HTTP
CNC	Windows Server 2016	10.0.146.141	445/tcp	SMB
CNC	Windows Server 2016	10.0.146.141	464/tcp	Kerberos
CNC	Windows Server 2016	10.0.146.141	593/tcp	HTTP RPC
CNC	Windows Server 2016	10.0.146.141	636/tcp	LDAPS
CNC	Windows Server 2016	10.0.146.141	3268/tcp	LDAP
CNC	Windows Server 2016	10.0.146.141	3269/tcp	LDAPS
CNC	Windows Server 2016	10.0.146.141	3306/tcp	MySQL
CNC	Windows Server 2016	10.0.146.141	3389/tcp	RDP
CNC	Windows Server 2016	10.0.146.141	5985/tcp	HTTP (WinRM)
Web Server	Centos 7	10.0.146.142	22/tcp	SSH
Web Server	Centos 7	10.0.146.142	25/tcp	SMTP

Web Server	Centos 7	10.0.146.142	80/tcp	HTTP
Web Server	Centos 7	10.0.146.142	110/tcp	POP3
Web Server	Centos 7	10.0.146.142	143/tcp	IMAP
Web Server	Centos 7	10.0.146.142	443/tcp	HTTP
Web Server	Centos 7	10.0.146.142	587/tcp	SMTP
Web Server	Centos 7	10.0.146.142	993/tcp	IMAP
Web Server	Centos 7	10.0.146.142	995/tcp	POP3
Web Server	Centos 7	10.0.146.142	2222/tcp	SSH
Web Server	Centos 7	10.0.146.142	8080/tcp	HTTP
Web Server	Centos 7	10.0.146.142	9002/tcp	HTTP
Traditional				
Public DB	OpenSUSE 15	10.0.146.143	22/tcp	SSH
Public DB	OpenSUSE 15	10.0.146.143	80/tcp	HTTP
Public DB	OpenSUSE 15	10.0.146.143	111/tcp	RPC Bind
Public DB	OpenSUSE 15	10.0.146.143	2049/tcp	NFS
Public DB	OpenSUSE 15	10.0.146.143	3306/tcp	MySQL
Public DB	OpenSUSE 15	10.0.146.143	8080/tcp	HTTP
Task	Windows Server 2022	10.0.146.144	80/tcp	HTTP
Task	Windows Server 2022	10.0.146.144	135/tcp	MS-RPC
Task	Windows Server 2022	10.0.146.144	139/tcp	NetBIOS-SSN
Task	Windows Server 2022	10.0.146.144	445/tcp	SMB
Task	Windows Server 2022	10.0.146.144	3389/tcp	RDP
Task	Windows Server 2022	10.0.146.144	5985/tcp	HTTP (WinRM)
AD DNS	Windows Server 2019	10.0.146.145	22/tcp	SSH
AD DNS	Windows Server 2019	10.0.146.145	53/tcp	DNS
AD DNS	Windows Server 2019	10.0.146.145	53/udp	DNS
AD DNS	Windows Server 2019	10.0.146.145	88/tcp	Kerberos
AD DNS	Windows Server 2019	10.0.146.145	135/tcp	MS-RPC
AD DNS	Windows Server 2019	10.0.146.145	139/tcp	NetBIOS-SSN

AD DNS	Windows Server 2019	10.0.146.145	389/tcp	LDAP
AD DNS	Windows Server 2019	10.0.146.145	445/tcp	SMB
AD DNS	Windows Server 2019	10.0.146.145	464/tcp	Kerberos
AD DNS	Windows Server 2019	10.0.146.145	593/tcp	HTTP RPC
AD DNS	Windows Server 2019	10.0.146.145	636/tcp	LDAPS
AD DNS	Windows Server 2019	10.0.146.145	3268/tcp	LDAP
AD DNS	Windows Server 2019	10.0.146.145	3269/tcp	LDAPS
AD DNS	Windows Server 2019	10.0.146.145	3389/tcp	RDP
AD DNS	Windows Server 2019	10.0.146.145	5985/tcp	HTTP (WinRM)

Network Diagram



Expected Services

Machine	Primary Services
ICS PLC	HTTP, Modbus, SSH
ICS CNC	SQL, HTTP, RDP, SMB
Web Server	HTTP, SSH
Public DB	SQL, SSH, NFS, SNMP
Task Box	HTTP, WinRM, SMB, RDP
AD / DNS	SSH, LDAP, Kerberos, RDP

Vulnerabilities and Mitigations

Host/System	Vulnerability	Mitigation(s)
10.0.146.140	The `bash` binary was copied to a binary called `csh` and given the setuid permission. This would allow anyone to run anything as root.	Would remove the `csh` binary.
10.0.146.140	User `Billie` has an empty password and is part of the sudo group. This allows anyone to login as `Billie` and then sudo to root.	Would set a password for `Billie`.
10.0.146.140	User `Josephine` has credentials in their `.bash_history`.	Would clear `Josephine's` bash history.
10.0.146.140	Option `PermitEmptyPasswords` is set to yes in the ssh_config. This allows users with empty passwords to log in with ssh.	Would set `PermitEmptyPasswords no`.
10.0.146.140	Anyone can run less with sudo without a password. This allows anyone to run any command as root.	Would remove the line `ALL ALL=NOPASSWD: /usr/bin/less` from `/etc/sudoers`.
10.0.146.140	The command `find` has the setuid permission allowing anyone to run any command as root.	Would run the command `chmod -s /usr/bin/find`.
10.0.146.140	The command `vim.basic` has the setuid permission allowing any user to run any command as root.	Would run the command `chmod -s /usr/bin/vim.basic`.
10.0.146.140	Every user is in the sudo group. This allows every user to run any command as root with their own password.	Would remove every user from the sudo group except for `test04`, `green03`, and `blueteam`.
10.0.146.140	Unnecessary http server is running on the machine. This could potentially be exploited by an attacker.	Would stop the Apache2 service and close port 80 using a firewall, such as UFW or iptables.
10.0.146.140	`Root` user has authorized SSH key called "Red Development", allowing direct access to `root`.	Would remove the key from `root's` authorized_keys file.
10.0.146.140	`Ubuntu` user has authorized SSH key called "Red Development", allowing direct access to `ubuntu`.	Would remove the key from `ubuntu's` authorized_keys file.

10.0.146.140	`Levi` user has authorized SSH key called "normaluser@legitplace", allowing direct access to `Levi` (potentially malicious key).	Would remove the key from `Levi's` authorized_keys file.
10.0.146.140	Hidden directory called `.secret_bad_guy` in the blueteam home directory.	Would remove the directory.
10.0.146.140	Socat binary installed at `/usr/bin/socat` and netcat installed at both `/usr/bin/netcat` and `/usr/bin/nc`. These allow a malicious user to set up a listener on the machine or connect to it using a reverse shell.	Would remove the binaries.
10.0.146.141	Guest (anonymous) authentication is enabled on this machine. While this isn't necessarily harmful, it makes enumeration of users, password policy, and shares possible without an account. This makes it easier for attackers to gain a foothold in the domain.	Would disable anonymous authentication by disabling the Guest account: `net user guest /active:no`
10.0.146.141	Null authentication is enabled on this machine. While this isn't necessarily harmful, it makes enumeration of users, password policy, and shares possible without an account. This makes it easier for attackers to gain a foothold in the domain.	Would set `HKLM\System\CurrentControlSet\Control\Lsa\RestrictAnonymous` to 0 to prevent null sessions.
10.0.146.141	Domain users `mike`, `user03`, `charlie`, `larry`, and `user01` on ventosa.energia all have extremely weak passwords.	Would temporarily disable these accounts and request that users change their passwords to meet a new, more strict, password policy.
10.0.146.141	The `user01` domain user kerberoastable, which means the hash of the account can be requested by any user. This is default for windows, but means that the password for the account must be long, complex, and random to ensure that users cannot crack the hash once requested. This user has a	Would change the account to have a more secure password. Usually, only machine and service accounts should be configured where the hash is requestable by any user and those accounts are configured with a randomly-generated 32-character password.

	very weak password that can be cracked in seconds. This user in particular is a domain administrator, which makes this a very dangerous vulnerability.	
10.0.146.141	The hash of the user `user01` is in `C:\Users\Public\Temp\kerb-Hash1.txt`. This indicates that a kerberoasting attack was done.	Would remove this file and reset the password `user01`.
10.0.146.141	The `mike`, `guest`, and `user01` users are asreproastable, which means any user can request the hash of this user and then crack it to obtain a password. This can sometimes have legitimate use cases, but rarely does. If it is legitimate, the user should have a long and complex password that cannot be cracked, which is not the case here.	Would run the following command in powershell if the user doesn't need pre-authentication to be off: `Get-ADUser -Identity <username> | Set-ADAccountControl -doesnotrequirepreauth $false` If pre-authentication being disabled is needed, we recommend requiring that the user creates a more complex password.
10.0.146.141	Attacker tools stored in `C:\Users\Charlie\Desktop\Rubeus.exe`, `C:\Users\Charlie\Desktop\Pspoof_c.exe`, `C:\Public\Temp\Rubeus.exe`, `C:\Public\Temp\Mimikatz.exe`. Rubeus and Mimikatz are tools that can be used for credential attacks in Windows. Pspoof_c appears to be a custom version of an exploit known as `Printspoofer`, which can be used to exploit a vulnerability to become Administrator.	Would remove these tools, as they have no legitimate or non-malicious use.
10.0.146.141	There is a potentially malicious binary, `netcat`, in `C:\Program Files\game data\pro device\mouse.exe`. This can be used by attackers to gain remote access to the machine or to exfiltrate files.	Would remove this program, as it was likely created by an attacker.

10.0.146.141	There is an unquoted service path for the service `PRODEVICE`, which runs `C:\Program Files\game data\pro device\mouse.exe` on boot as an Administrator. Because `C:\Program Files\game data` is writable by all users, this can allow an attacker to escalate privileges to Administrator.	Would either remove this service and make the path `C:\Program Files\game data` only writable by an Administrator, or would add quotes in the service path.
10.0.146.141	Encryption was disabled in the SMB server configuration. This could allow attackers to intercept sensitive data.	Would enable encryption with the powershell command `Set-SmbServerConfiguration –EncryptData $true`.
10.0.146.141	SMBv1 is enabled, which can allow attackers to read or even modify files and commands over the network.	Would disable SMBv1 with the powershell command `Disable-WindowsOptionalFeature -online -FeatureName SMB1Protocol`.
10.0.146.141	Weak password policy that has a 7 character password length requirement and no lockout policy for repeated login failures.	Would implement a password policy that will require a minimum of 15 characters including special characters. Also would ensure that a user will get locked out after 3 incorrect logins.
10.0.146.142	CVE-2021–3156 - Version 1.8.23 of `sudo` allows privilege escalation.	Would upgrade `sudo` to the most recent version.
10.0.146.142	Every user except `test04` is part of the wheel group. This allows every user to run any command as root using sudo.	Would remove every user from the wheel group except for the users `blueteam` and `green03`.
10.0.146.142	Any user in the `wheel` group can run any command as root without a password.	Would configure `sudo` to require a password from users in the `wheel` group.
10.0.146.142	Socat binary is installed at `/usr/bin/socat` and the netcat binary is installed at `/usr/bin/netcat`, `/usr/bin/nc`, and `/usr/bin/ncat`. These allow a malicious user to set up a listener on the machine or connect to it using a reverse shell.	Would remove the binaries.
10.0.146.142	Telnet is configured. Telnet is a protocol with no encryption, opening communications up to man-in-the-middle attacks and passive sniffing of credentials.	Would disable the Telnet protocol. A suggested alternative is Secure Shell (SSH), which can be configured with asymmetric encryption and also provides authentication, mitigating man-in-the-middle attacks.

10.0.146.142	CVE-2020-14367 - a vulnerable version of chronyd.	Would upgrade the version of chronyd or remove chronyd from the system.
10.0.146.142	Root login through SSH allowed. Although this might not be a critical issue on its own, it leaves the system more vulnerable than necessary, because most brute force attacks take place against the `root` account.	Would add the line `PermitRootLogin NO` to the SSH configuration file.
10.0.146.142	SSH configuration option `PermitEmptyPasswords` commented out. If a user's password is inadvertently set to the empty password, attackers will be able to log in as them without a password.	Would set `PermitEmptyPasswords NO` in the SSH configuration file.
10.0.146.142	`Root` user has authorized SSH key called "Red Development", allowing direct access to root.	Would remove the key from root's authorized_keys file.
10.0.146.142	`Centos` user has authorized SSH key called "Red Development", allowing direct access to root.	Would remove the key from `Centos's` authorized_keys file.
10.0.146.142	`Levi` user has authorized SSH key called "normaluser@legitplace", allowing direct access to Levi (potentially malicious key).	Would remove the key from `Levi's` authorized_keys file.
10.0.146.142	`Green03` user has an authorized SSH key in its directory (potentially malicious).	Would remove the key from `Green03's` authorized_keys file.
10.0.146.143	A custom service file called "dbus-org-manager" was launching a socat listener at boot as root on port 8080. This allowed anyone to connect to the machine and run any command as root.	Remove the malicious service file and stop the malicious service.
10.0.146.143	The root's crontab had an entry that was launching a socat listener on port 80 once every hour. This allowed anyone to connect to the machine and run any command as root.	Remove the malicious cronjob.
10.0.146.143	The binary `cp` has the setuid permission. This ultimately allows attackers to escalate	Run the command `chmod -s /usr/bin/cp`.

	from standard user privileges to root command execution, as well as read and write to any file on the system.	
10.0.146.143	Anyone can run `find` as root without a password. This allows for anyone to run any command as root.	Remove the line `ALL ALL=NOPASSWD: /usr/bin/find` from the `/etc/sudoers` file.
10.0.146.143	Every user other than `kandy` is part of the wheel group. This allows every user to run any command as root using sudo	Remove all users from the wheel group except for users `green03`, `test04`, and `blueteam`.
10.0.146.143	Plaintext credit card info is stored in the database. This allows anyone with access to the database to steal our customer's passwords. Per section 3.5.1 of the PCI-DSS standards, such information must be "rendered unreadable anywhere it is stored," which is most commonly achieved using some type of encryption or hashing.	Either encrypt the cardholder information using strong reversible encryption like AES or use one-way hashing to verify to render it unreadable.
10.0.146.143	User `nobody` (a default account on Linux systems) has been given interactive shell capabilities in `/etc/passwd`. This often serves as a persistence strategy for malicious actors, given that the account does not normally have access to a terminal shell.	Replace `nobody:x:65534:65534:nobody:/var/lib/nobody:/bin/bash` with `nobody:x:65534:65534:nobody:/var/lib/nobody:/usr/bin/nologin` in the `/etc/passwd` file.
10.0.146.143	`Levi` user has authorized SSH key called "normaluser@legitplace", allowing direct access to Levi (potentially malicious key)	Remove the key from `Levi's` authorized_keys file.
10.0.146.143	The `PermitRootLogin` option is enabled for the SSH service, which acts as a single point of failure in case of a breach, and eliminates audit trails indicating which system administrator performed which actions.	Change `PermitRootLogin yes` to `PermitRootLogin no` in `/etc/ssh/sshd_config.`
10.0.146.143	The entire filesystem is shared on NFS in insecure mode with reading and	Remove the option `insecure` from the `/etc/exports` file.

	writing capabilities. This may allow a savvy attacker to modify any file on the machine.	
10.0.146.143	SNMP is configured to use `mysecret` as the secret community string with read and write attributes. This is an easily guessable string and would allow anyone to view and modify configurations on the machine.	Change the read/write community string to be a more complex string.
10.0.146.143	SNMP is configured with the default community string `public`. While this is not directly harmful, it allows for anyone to view the machine configurations without logging in.	Remove the line `rocommunity public` from the `/etc/snmp/snmpd.conf` file.
10.0.146.143	SNMP is configured to run `/bin/bash` whenever anyone connects to SNMP. This could allow anyone with write access to SNMP to add arguments to it and run any command as root.	Remove the line `extend test /bin/bash` from the `/etc/snmp/snmpd.conf` file.
10.0.146.143	The users that can access MySQL remotely have the weak guessable password: `password`. This could allow anyone to log into the MySQL database remotely and have admin control over it.	Change the `root`, `sqlUser`, and `sqlAdmin` passwords in MySQL to be more complex.
10.0.146.144	The firewall is disabled on Windows 2022. This is generally poor practice as it allows an attacker to more easily run listeners and may unwittingly open internal processes for attack.	Turn on the firewall.
10.0.146.144	A group policy is configured to disable antivirus. While this will eliminate false positives, this will also open up the computer to attack by maliciously-installed components such as command and control beacons and malware.	Enable Windows Defender by opening gpedit.msc and going to Computer Configuration > Administrative Templates > Windows Components > Microsoft Defender Antivirus > Real-time Protection. Then, disable “Turn off real-time protection.”
10.0.146.144	Guest (anonymous) authentication is enabled on this machine. While this isn’t necessarily harmful, it makes	Disable the Guest password from Local Users and Groups.

	enumeration of users, password policy, and shares possible without an account. This makes it easier for attackers to gain a foothold in the domain.	
10.0.146.144	Global read/write access on `replication` SMB share. This makes it possible for an attacker to steal, modify, or even remove backups, which can be detrimental in the case that backups are needed in an emergency. It is also possible for an attacker to use this in a relay attack in which a malicious LNK file can be uploaded to the SMB share. If a user browses to the share with a malicious LNK file, an attacker can steal that user's NetNTLM hash and attempt to crack it.	Ensure that only domain users can read and write to this share by changing permissions in the Server Manager.
10.0.146.144	Blank password allowed for RDP. This makes it easier to enumerate domain information on the machine, but also makes it possible for the attacker to exploit misconfigurations like that of the stickykeys and utilman misconfigurations mentioned below.	Set the `LimitBlankPasswordUse` registry key to 1 in `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa`.
10.0.146.144	Activating Sticky Keys by pressing Shift 5 times prior to logging in opens up a command prompt as NT AUTHORITY/SYSTEM. This allows attackers immediate administrative access to a machine without credentials.	Remove the `Debugger` entry in `HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\sethc.exe`.
10.0.146.144	Opening up the Utility Manager (the "Accessibility" or "Ease of Access" icon) prior to logging in opens up a command prompt as NT AUTHORITY/SYSTEM. This allows attackers immediate administrative access to a machine without credentials.	Remove the `Debugger` entry in `HKLM\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Image File Execution Options\utilman.exe`.
10.0.146.144	Admin shell through `DisplaySwitch` (switch displays shortcut) on login.	Remove the `Debugger` entry in `HKLM\SOFTWARE\Microsoft\Windows`

	This allows attackers immediate administrative access to a machine without credentials.	`NT\CurrentVersion\Image File Execution Options\sethc.exe`.
10.0.146.144	`hacker` user, which is a local administrator, with a very weak password. This user is potentially an artifact of an attacker who was already on the system. It can be used to log in by the previous attackers and may even be used by future attackers as a result of the easily guessable password.	Remove this user, as it almost definitely belongs to an attacker.
10.0.146.144	Users `larry` and `mary` have weak passwords for their local accounts. These passwords are also reused for their domain accounts.	Change the passwords of users `larry` and `mary`.
10.0.146.144	An administrator set the password for the `test04` user directly in the command line, which means powershell history can be used to obtain the user's password.	Change this password and ensure that, in the future, passwords are not stored to logs if possible.
10.0.146.144	An administrative user downloaded the files `mm.exe`, `mm.sys`, `mimilib.dll`, and `mimispool.dll`. This can be seen in the PowerShell command history. All of these are instances of a malicious tool known as Mimikatz, which attackers often use to steal credentials on Windows systems. This could indicate that an attacker had administrative access on this machine.	Remove these files, as they are only used for malicious purposes.
10.0.146.144	Encryption was disabled in the SMB server configuration. This could allow attackers to intercept traffic.	Enable encryption with the powershell command `Set-SmbServerConfiguration –EncryptData $true`.
10.0.146.144	Data sent through SMB was not signed, allowing for man-in-the-middle modification of data.	Enable security signature with the powershell command `Set-SmbServerConfiguration –EnableSecuritySignature $true`.
10.0.146.144	The `mary` user has credentials in their account description. This description can be viewed by anyone	Remove the description of the account and temporarily disable this account.

	authenticated to the domain, which makes it easy for attackers to find.	
10.0.146.144	Weak password policy that has a 7 character password length requirement and no lockout policy for repeated login failures.	Implement a password policy that will require a minimum of 15 characters including special characters. Also would ensure that a user will get locked out after 3 incorrect logins.
10.0.146.145	xampp is installed and runs many vulnerable services that allow command execution as an admin. None of the services in xampp are necessary for this host. This includes a file called `info.php` and `info_old.php`.	Uninstall XAMPP.
10.0.146.145	PHP application on xampp does not disable any dangerous functions. This can make exploitation more likely in the case of a vulnerability in the code.	Uninstall XAMPP and replace it with a more secure alternative. XAMPP was designed for development and testing only and is not for production use. If this cannot be done, manually disable dangerous functions in XAMPP.
10.0.146.145	Users `Lisa` and `Brian` have local administrative privileges when they don't need to. It is considered best practice to not have accounts that will be logged in on a daily basis as a local administrator.	Remove `Lisa` and `Brian` from the Builtin\Administrators group.
10.0.146.145	Unnecessary service tftp is installed and running. This could be a security risk, as tftp is unauthenticated, unencrypted, and has no concept of file permissions (meaning that attackers can access any public file on the system).	Remove tftp from the machine.
10.0.146.145	Attacking tools `Handlekatz.exe` and `Rubeus_C.exe` are installed on the machine. Both of these tools can be used for obtaining and abusing credentials in Windows.	Remove the unnecessary tools. These tools can allow attackers to more easily conduct their attacks.
10.0.146.145	The domain controller NTDS file, `ntds.dit`, which includes domain information - including user passwords -	Remove this file, as it should not exist directly in System32.

	seems to have been copied to `C:\Windows\System32`, which is not the appropriate location. This indicates that an attacker may have attempted to exfiltrate it.	
10.0.146.145	The `larry` user has social security numbers of employees on his desktop. This file was created by the `mary` user, which could indicate lateral movement between accounts. Per section 3.5.1 of the PCI-DSS standards, such information must be "rendered unreadable anywhere it is stored," which is most commonly achieved using some type of encryption or hashing.	Either encrypt the cardholder information using strong reversible encryption like AES or use one-way hashing to verify to render it unreadable.
10.0.146.145	The "Public" user has a file with credit card information in its Documents folder and has a copy of the file with social security numbers from the `larry` user. Per section 3.5.1 of the PCI-DSS standards, such information must be "rendered unreadable anywhere it is stored," which is most commonly achieved using some type of encryption or hashing.	Either encrypt the cardholder information using strong reversible encryption like AES or use one-way hashing to verify to render it unreadable.
10.0.146.145	The `Public` user has a file called `PowerView.ps1` in its Documents folder. PowerView is a tool commonly used by attackers. This file was created by the Administrator user, which indicates that an attacker likely was able to escalate privileges on the machine.	Remove PowerView from the machine, as it is not used for any legitimate purpose.
10.0.146.145	Encryption was disabled in the SMB server configuration. This could allow attackers to intercept traffic.	Enabled encryption with the powershell command `Set-SmbServerConfiguration –EncryptData $true`.
10.0.146.145	Weak password policy that has a 7 character password length requirement and no	Would implement a password policy that will require a minimum of 15 characters including special

	lockout policy for repeated login failures.	characters. Also would ensure that a user will get locked out after 3 incorrect logins.

System Hardening

The team took a multifaceted approach in hardening Energia Ventosa's infrastructure to long-term reliability and protection against compromises in confidentiality, integrity, and availability. Leveraging a defense-in-depth strategy, the company's infrastructure has multiple layers of defense that will keep it secure in the event attackers breach a security control.

On the host level, we configured weekly system and software updates. This will ensure that no component of the network will remain out-of-date and vulnerable to new exploits. Additionally, this change reduces the time required to maintain the network.

Next, the team audited existing accounts and enforced a policy of least privilege across the infrastructure. If an account was not explicitly required, it was disabled. Otherwise, the account's privileges were minimized to ensure it could still carry out required tasks while reducing the risk of unauthorized access and abuse. For Active Directory, the team used BloodHound and ADMiner to map the complex relationships, identify attack paths, and subsequently remove unnecessary permissions and group memberships. This limits privilege escalation attacks and lateral movement in the event an attacker breaches an account. It is important to note that reduced permissions may introduce friction for users performing job-related tasks, but it is a worthwhile trade-off for well-defined security management and a strong security posture.

Following the principle of least privilege, non-essential ports on the machines were closed. Using netstat and the Network Mapper (Nmap) tool, each machine was scanned and open ports were audited to determine if there was a business need. Unnecessary ports were closed using the host-based firewall and disabling the unneeded services on each host. This step reduces the attack surface of the network and reduces the necessary monitoring workload.

Anti-malware software was installed on all traditional machines. While not a catch-all, it does identify common threats adding an extra defensive layer. For Windows computers, the default anti-malware software Windows Defender was used and supplemented with Malwarebytes, as it has a strong virus signature database.

For each host on the network, we took measures to ensure that all necessary services were hardened following industry standards. Examples include disabling SSH root authentication, enabling NFS secure mode, restricting SMB share access, and ensuring that only the explicitly required users had access to MySQL. By hardening our services, we further reduced the attack surface and potential for abuse.

On the Windows machines, PowerSTIG was used in order to apply Desired State Configurations (DSCs) that would ensure compliance with DISA Security Technical Implementation Guides (STIGs). This ensures compliance with Department of Defense standards, which are robust. Group Policies were also created and enforced, where allowed, to guarantee that user passwords are strong and that secure defaults are applied. For example, this included enabling SMB signing, enforcing strong encryption types for Kerberos tickets, and activating Windows Defender active scanning.

Finally, the team established endpoint logging, auditing, and detection tools in conjunction with a layer of network intrusion detection and prevention. For the endpoint logging and auditing, we

leveraged Sysmon and auditd to form a strong foundation of host-based logging. Sysmon, in conjunction with open source security-focused configuration from SwiftOnSecurity, enables us to collect enriched security logs from the Windows machines. These logs are more descriptive than the standard logs that get generated by the operating system in its default state. With this, we are able to more quickly identify the source of problems, the types of traffic that they came from, and their effect on the overall security posture in the case of a compromise. Similarly, on Linux, we utilized the auditd logging utility in combination with the well-known configuration Florian Roth security ruleset to generate enhanced logs on all Linux-based hosts.

Logging, however, is cumbersome and incomplete without a way to centrally store and analyze individual logs. Collecting evidence from each machine every time a network investigation is conducted is simply unreasonable and time-consuming. The team solved this by installing Wazuh on all of the machines in the network to manage our system auditing from a single centralized location. Not only are we then able to aggregate all of the events occurring on the network, but we are also able to visualize the data with graphs, charts, and other visualization tools that come bundled with Wazuh, which has a backend built on the Elastic Search software. This results in a convenient way to monitor the various different events going on in the Energia Ventosa network, with an easy way to interpret and take action based on trends across all machines.

Finally, we incorporated the Suricata network analysis and threat detection software to ensure live network monitoring. Although this can be achieved with tools such as Wireshark, Suricata offers a long-term solution that records and analyzes network traffic across larger periods of time, comparing traffic to known malicious interactions that may raise flags for system and network administrators. This functions in addition to the local system-level logging we have set up because attackers have to first traverse the network before being able to carry out their techniques and procedures on local machines. As such, a lot of attacks and offensive techniques can be stopped at the network level and mitigated entirely without having to rely on individual defense systems on a per-machine basis. With network security monitoring on top of all the measures mentioned before, Energia Ventosa can be assured that its digital infrastructure has an up-to-date security posture that will last and outlive many attack patterns and types.

Tools List

- Malwarebytes
- Nmap
- Wireshark
- PEASSng
- Sysmon
- Auditd
- Wazuh
- Netexec
- Autoruns
- ProcessExplorer
- BloodHound
- ADMiner
- Certipy
- Nessus
- Suricata
- Sysdig