National Cyber Security Centre
a part of GCHQ

Communications Security Establishment
Canadian Centre for Cyber Security

Centre de la sécurité des télécommunications
Centre canadien pour la cybersécurité

Australian Government
Australian Signals Directorate

ASD AUSTRALIAN SIGNALS DIRECTORATE
ACSC Australian Cyber Security Centre

Te Tira Tiaki
Government Communications Security Bureau

National Cyber Security Centre
PART OF THE GCSB

Guidance on digital forensics and protective monitoring specifications for producers of network devices and appliances

This guidance has been developed with contributions from partnering agencies and is included in a series of publications aiming to draw attention to the importance of edge device cyber security measures.

It is produced by the UK National Cyber Security Centre (NCSC) in partnership with the Australian Signals Directorate (ASD), US Cybersecurity and Infrastructure Security Agency (CISA), the Canadian Centre for Cyber Security – part of the Communications Security Establishment (CSE), the US Federal Bureau of Investigation (FBI), and New Zealand's National Cyber Security Centre (NCSC-NZ).

Context to this guidance

As the number of malicious actors and their capabilities against critical and systemically important infrastructure increases, so does the number of compromises of network devices and appliances. Previous compromises have affected both physical and virtual network devices, such as edge perimeter security solutions and routers, as well as network attached storage.

Network devices and appliances are prime targets for malicious actors because they play a crucial role managing and processing traffic. When targeting these devices, malicious actors have exploited vulnerabilities and insecure design features to gain and maintain valuable accesses. These actors can remain inside networks until detected and denied access.

These devices and appliances can be targeted when they lack secure by design/default aspects, regular firmware updates, or if they have weak authentication measures and provide limited logging, making it difficult to detect suspicious activity. Additionally they may not be configured securely, lack proper network segmentation and use unsupported, or end-of-life (EOL) hardware, increasing their vulnerability to attacks.

What and who is this guidance for?

This guidance outlines expectations for the **minimum requirement for forensic visibility**, to help network defenders secure organisational networks both before and after a compromise. **Network defenders** are encouraged to consider these features when selecting new physical and virtual network devices.

Device manufacturers are encouraged to include and enable standard logging and forensic features that are robust and secure by default, so that network defenders can more easily detect malicious activity and investigate following an intrusion.

By following the minimum levels of observability and digital forensics baselines outlined in this guidance, **device manufacturers** and their customers will be better equipped to detect and identify malicious activity against their solutions. Device manufacturers should also use it to establish a baseline of standard features to include in the architecture of network devices and appliances, to facilitate forensic analysis for network defenders.

Logging requirements

Where possible, putting in place the features below as minimum requirements will support threat detection and response for physical and virtual network devices.

Devices and appliances should support the logging / recording of events related to:

- authentication including username, method used (such as password, SSH key, certificate, multi-factor authentication), source IP address or hostname and relevant session identifiers for both successful and failed attempts
- technical support events and interactions that rely on tools from vendors requiring authentication using a specialised licence key from the network device manufacturer
- service and application-level logging, such as HTTP/HTTPS requests and responses, interactive shell sessions (CLI) and similar services, including client IP addresses, request methods, URIs accessed, protocol versions, user agents, session identifiers, response status codes and the amount of data transferred
- process creation including parent process, executable path, username and arguments
- log the process exit code and termination reason (such as normal exit, crash, killed by signal)
- dynamic loading and unloading of modules and libraries, including the module or library name, version, file path, associated process ID and user context
- file system creation, modification and deletion that could facilitate post- breach investigations, especially in critical directories such as the web root, configuration directories and system binaries
- DNS queries performed including A, AAAA, TXT, SOA, NS, MX and PTR records, their responses, the querying process and the destination DNS server
- firmware or software updates, including both successful and failed attempts, current and target version numbers, update source (such as URL or repository), digital signatures or checksums verified, user or process initiating the update and the full error messages encountered
- configuration changes of a device, especially if connected directly to the internet (see the US binding operational directive (BOD) 23-02 for more about the risks associated with internet exposed interfaces), logging parameters changed, previous and new values, user or process making the change and method of change (such as CLI, API, web interface)
- configuration backup, export and/or download operations
- log attempts to clear, rotate, or manipulate log files
- diagnostic, recovery and safety events
- peripheral insertions, such as USB, MMC, SDCard, etc

Note: To prevent local data storage issues, it is recommended that organisations use appropriate remote logging solutions where feasible, taking into account bandwidth requirements, but the recommendations here can also be considered for local or remote logging configurations.

Secure logging actions

If an incident occurs, secure logging will help investigations. Putting in place the recommended actions below will support this work.

- Logs should be collected in an easily ingestible format that can be imported for forensic analysis into security information and event management (SIEM) tools. All timestamps with significant events, such as device boots and reboots, should be in ISO 8601 format, including milliseconds, and set to the coordinated universal time standard (UTC), for example: 2024-06-27T14:58:43.085872Z.
- All systems should run the network time protocol (NTP) from reputable and redundant sources and alert loudly when there is an NTP failure, so that network defenders know to investigate and remediate.
- Logs should contain a Globally Unique ID (GUID) for every device.
- Sufficient local storage capacity should be available before log rollover occurs, to be of maximum benefit.

Remote logging / Event push support

Devices and appliances should support near-real-time log transfer using a standards- based protocol, protected using transport layer security (TLS) encryption in a recognised secure configuration. Log formats should be fully documented to allow third-party platforms and tools to ingest them and be machine readable using a standardised format.

Devices and applications should include by default standard features that serve to generate keep-alive messages (heartbeat, as evidence they are operating correctly even when there are no log-generating events). These heartbeat messages should include:

- version number
- SHA256 digest of current configuration
- last reboot date and time of the device
- its Globally Unique ID (GUID)

Network devices and applications should come packaged with remote logging security features by default and, where possible, management planes segregated. Devices and applications should warn network administrators if remote logging is disabled or misconfigured through the administration interface.

Forensic data acquisition requirements

Volatile data collection

Through a local or remote privileged interface, devices and applications should support volatile collection of the current running state of the device or appliance.

This volatile data collection feature should trigger a log generation to help facilitate automatic analysis, as well as human investigators to detect anomalous events.

Collecting this information may also help a malicious actor as much as defenders, so it's important to build this feature in a secure way.

Where it is supported by the operating system, volatile data logging should support collection of:

- process information including parent/child relationships and arguments
- process memory maps
- process dynamically loaded modules, including path
- process handle information, including files and sockets
- process environment variables
- memory both at a kernel and individual process level
- firewall and/or packet processing rules
- mounted filesystems
- network connections, including source and destination addresses, ports and protocols
- current Address Resolution Protocol (ARP) entries mapping IP addresses to MAC addresses
- For network switches - Content Addressable Memory (CAM) tables listing MAC addresses associated with specific switch ports
- Dynamic Host Configuration Protocol (DHCP) lease tables, including client MAC addresses, assigned IP addresses, lease durations and options provided
- active administrative and user sessions, regardless of interface and transport
- kernel dynamically loaded modules
- volatile filesystem contents, such as. /tmp
- raw volatile filesystems
- crash/core dumps
- all available logs available from a system and application level

If information needs to be redacted as part of volatile collection to protect a vendor's intellectual property, their security or the integrity of a services security posture, this redaction should be obvious, for example, by marking it as *REDACTED*, in the collective artefacts and appropriately documented.

Non-volatile data collection

Devices and appliances should support full non-volatile storage collection of the entire data storage capability of the device, ideally through standard interfaces. A system owner should be able to decrypt the contents of the stored data, potentially involving vendor support, to inspect it with standard tools where possible and where the security risks of being able to do is managed.

Initial configuration of the system may be required to make this possible, for example 'bring your own key'. It is recommended that protection of keys be a primary consideration. Additionally, for physical devices, the device's firmware and hardware should be designed to prevent unauthorised data extraction, such as implementing secure boot processes, Trusted Platform Module (TPM) integration and disabling unnecessary physical interfaces that could be exploited. Any interfaces used for non-volatile data collection should require strong authentication and authorisation controls to prevent misuse.

Further resources

Secure by design guidance

This guidance supports wider secure by design principles. For more on this, see:

- NCSC secure design principles
- CISA guidance, including a CISA pledge
- ASD secure by design principles

Other guidance

- NCSC guidance on logging and protective monitoring
- ASD guidance on best practices for event logging and threat detection
- CCCS guidance on network security logging and monitoring