

On-site data collection agent / DCA

Security and operations information

Technical reference for IT approval and operation

Document version: 1.21 · Reviewed: 25 October 2026

For: IT operations, network security, endpoint protection and data protection

Publisher: Sziráki Márton e.v. / PrinterMS · Technical description; no separate SLA commitment.

The PrinterMS agent reads printer identifiers, meters, supply levels and status, then forwards them to the service provider's PrinterMS account. This document sets out the technical conditions needed to approve installation and verify operation.

1. Printers SNMP / IPP / IPPS Targeted read-only queries	→	2. Local agent Validation, local outbox Scheduled forwarding	→	3. PrinterMS API HTTPS / TCP 443 Storage and authorised viewing
Print content	Does not query documents, print jobs, print queues, address books or user files; does not submit print jobs or change printer settings.			
Network access	Outbound HTTPS to <code>api.printerms.com</code> ; LAN SNMP/IPP. Periodic discovery and local mDNS are enabled by default. There is no agent server accepting remote sign-ins.			
Remote control	Responses to outbound HTTPS requests may carry configuration and defined operational instructions. There is no general remote command line or operation for running arbitrary scripts.			
Local protection	Isolated collection service on Windows, Linux and macOS. Access to the agent's files and credentials is restricted; the token is DPAPI-protected on Windows (E1), the outbox has no separate application-level encryption.			
Effect of stopping	The direct printing path keeps working. New technical observations and alerts based on them pause; historical data stored in the cloud remains.			

This description covers the functionality available at the review date. The download page, agent version display and installation record identify the actual release version and SHA-256 hash. Pages 6 and 9 explain differences between platforms and installation methods.

Data and data minimisation	2	Purpose of technical metadata	3
Network connections	4	Protocols and discovery	5
Installation and permissions	6	Local storage and outages	7
Cloud, access and retention	8	Activation and remote operations	9
Threats and safeguards	10	Verification, stopping and contact	14

Data collected and transmitted

Measurement packages contain technical device data. Separate metadata about the host and agent operation is needed for activation, troubleshooting and connection monitoring. Some fields below are only present when supplied by the device or used by the relevant feature.

Data category	Exact content and purpose	Processing location
Printer identifiers	Serial number, manufacturer, model, printer IP address and available firmware version, to match measurements to a device. For a local USB connection, the loopback address is not submitted as the printer's IP address.	Locally and in the PrinterMS cloud.
Meters	Total, monochrome, colour and scan meters, and the total meter's unit; measurement time, sometimes device uptime and a meter-reset indicator.	Local outbox, then cloud.
Supplies	Supply name, class, colour, percentage or raw level, capacity, unit and status.	Local outbox, then cloud.
Status and errors	Error bits, active error codes, severity, source and sanitised manufacturer error description; measurement time, sometimes total page count.	Local outbox, then cloud.
Readability indicators	Success, source and fixed reason code for meter, toner and error reads, and the version of the query definition used.	Local outbox, then cloud.
Device-support walk	On explicit request, or if the organisation's Owner has opted into the device-support programme (see 08): a one-off, redacted walk of a printer's built-in printer-related MIB branches, so PrinterMS can add support for a printer not yet fully readable. Optionally, a meter-sheet or display photo attached by the requester.	Redacted locally, then cloud; 12 months.
Trial-rule read	Only if PrinterMS publishes a signed trial package: during the daily meter read, the agent also reads the same printer's meters using the trial rule. The result is not displayed, billed or used to overwrite the live value; it is only used to validate the rule.	Cloud; 30 days.
Activation metadata	Host name, platform, processor architecture, agent version and the private IPv4 subnets seen by the host (up to 16). The server also records the connection's source IP address.	Activation process and cloud agent record.
Agent status	Version, platform, architecture, connection time, device count; pending and rejected package counts, outbox size, fullness; update status, fixed error code and a fixed-list reason (e.g. name resolution, certificate or locked file), without URLs, paths or raw error text; IP addresses and ranges set in the local <code>agent.json</code> file (up to 64); cap mode, skipped cloud ranges, out-of-scope printers, update mode, SNMP minimum version, weak-crypto opt-in.	Status reporting to the cloud; local operational state.
Local technical state	Known printer routes and system descriptions, SNMP version, previous meter values, scheduled measurements, configuration and logs.	The agent's own local files. Their full contents are not uploaded automatically.
Credentials	Activation secrets and the agent's device token; locally configured SNMP community or SNMPv3 username and keys. These are not measurement data.	Activation/API authentication and local SNMP respectively. SNMP credentials are not part of cloud telemetry.

Content outside the scope of collection

The agent does not read the content or history of print, scan, copy or fax jobs, spool files, filenames, address books, emails, browsing history or keystrokes. It does not query user sessions or take screenshots. It reads and writes its own configuration and files.

Support data transfer: the agent does not automatically upload the full local log. A support request may include the version, OS, time, error code and a reviewed log excerpt. Only share secrets or the full data directory through a separately agreed protected channel.

Metadata that may identify a person. A host name, IP address or free-text printer field may identify a person in the customer's environment. We therefore do not claim that all technical data is necessarily non-personal. Sanitised error text is not automatically anonymised either; device names without personal names are advisable.

Why is this data needed?

Alongside printer measurements, we handle technical data to identify installations, select compatible software and investigate operational faults. The table below shows which operational task each data item supports.

Data	Purpose and interpretation
Agent host name (hostname)	The activation approver and operator can identify which computer hosts the agent. With multiple installations, it helps select the right host before troubleshooting or remote maintenance. The host name is neither a username nor a credential.
Connection source IP address	Records the network source seen by the server. During a connection or incident investigation, it can be compared with the customer's firewall and proxy logs. With NAT or a proxy, several hosts may share an egress address; it is not necessarily the host's internal IP address and does not reliably identify a host on its own. During activation, the approver also sees whether the address matches that of an existing agent in the organisation.
Printer's internal IP address	The local agent uses this address to reach the printer. In the cloud device record, it helps IT locate the affected device and investigate address changes or connectivity issues. Knowing the IP address does not give the PrinterMS cloud direct access to the customer's network.
Platform and processor architecture	The Windows/Linux/macOS platform and architecture, such as amd64/arm64, determine which agent package is compatible and which maintenance method is supported. They do not describe the processor's serial number or performance, or the employee's work.
Agent version and update status	Used to check the supported release and features, select an update target and distinguish successful, failed or rolled-back updates. In a support request, it helps associate a problem with the right software version.
Serial number, manufacturer, model and firmware	The serial number helps match a measurement to the correct printer; the manufacturer and model help select the correct reading rule. The available firmware version can be used to investigate compatibility and service issues. Reading firmware information does not confer permission or functionality to update printer firmware.
Timestamps, device count and outbox status	The latest connection and measurement times indicate data freshness. The device count helps verify discovery; pending/rejected package counts and outbox size help detect stalled delivery. These are technical operational indicators, not the contents of printed documents.
Configured IP addresses and ranges	Define the monitored printers and allocate discovery work. Local subnets reported during activation are only suggested as the address ranges for a new network; the server drops suggestions the cloud may not set, and the approver can change the rest. The local cap mode for the cloud address list, the skipped ranges and the number of printers outside the scope show on the network page which part of the cloud address list does not apply. Multicast discovery has a separate setting; the customer can also enforce actual destination restrictions through firewall rules.

What should IT approval record?

Use host and printer names without personal names; define monitored addresses, collection categories and staff who may access the data. IP addresses and hostnames may relate to individuals in the customer's environment. Disabling collection categories does not automatically disable activation and connection metadata; there are currently no separate per-field controls for these.

Printer measurements, host metadata and credentials stored only locally are separate data categories. The data inventory describes where they are stored; the cloud data processing section explains access and retention.

Network connections and endpoints

The agent does not provide a server accepting inbound application connections, a web management interface or a remote desktop. Connections originate from the agent; discovery also opens short-lived sockets to receive UDP responses. No port forwarding or internet-facing management port is needed.

Destination / direction	Port, protocol	Purpose and enabling condition
api.printerms.com Outbound, internet	TCP 443 HTTPS	Normal cloud connection. Activation, telemetry, status reporting, configuration, token management, signed definitions and the update manifest. If the update file cannot be downloaded from agent.printerms.com, the agent (from 1.9.0) makes one attempt to fetch the same file through this address using its device token.
Approved printer IPs Outbound, local/routed network	UDP 161 SNMP	Primary identification and reading. Also probes discovery address candidates, which may include devices other than printers.
Approved printer IPs Outbound, local/routed network	TCP 631 IPP / IPPS	Identification and fallback queries, enabled by default. Can be fully disabled locally.
224.0.0.251 ff02::fb Local multicast and responses	UDP 5353 mDNS / DNS-SD	Time-limited searches for _ipp._tcp and _ipps._tcp, enabled by default. Can be disabled separately; scanless mode also disables them.
239.255.255.250 ff02::c Local multicast and responses	UDP 3702 WS-Discovery	Optional, disabled by default. Requires an enabling local or cloud setting; scanless mode also disables it.
127.0.0.1 :::1 Host-local only	TCP 60000-65535 HTTP / IPP	Optional IPP-over-USB connector, disabled by default. Queries the separately installed OpenPrinting ipp-usb service; this is not a general local port scan.
Host's configured name resolvers OS service	As specified by local DNS policy	To resolve cloud addresses. The agent does not require a particular public DNS server. HTTPS connectivity may use IPv4 or IPv6.

Installation and updates: a separate approval scope

The public download page is available over HTTPS at agent.printerms.com. The public page, installer and command contain no secret, and no PrinterMS sign-in is needed on the installation host. Release files and command-based installation may use github.com/PrinterMS/agent-releases, with redirects to a GitHub-designated download host. The short installation commands (curl -fsSL https://agent.printerms.com/install.sh | sudo sh and irm https://agent.printerms.com/install.ps1 | iex) fetch a small bootstrap script from agent.printerms.com. It runs the pinned release installer as administrator only after verifying the SHA-256 hash (and, on Windows, the PrinterMS digital signature). The script consists of a single function called on its final line, so a truncated download runs nothing. The download page also provides the longer, hash-pinned “Verified installation” command. Docker also needs access to the selected image registry (usually gcr.io). Windows Setup only finalises installation after approved activation; on Linux, macOS and Docker, the service is installed first and waits without discovery or data submission until approval.

Install link: the link opens the agent.printerms.com/t#TOKEN page; the token is in the part after #, so the browser does not send it when requesting the page. Windows Setup and the macOS package read it from their own file name (PrinterMS-Agent-Setup-TOKEN.exe), install.sh from the DCA_INSTALL_TOKEN_FILE file (deleted at start) or the DCA_INSTALL_TOKEN variable, so treat the link, the file and the command as secrets until expiry. After a failed link installation only the token and a fixed support code (for example PMS-S001) are sent to api.printerms.com.

During Windows installation/updates, the operating system checks the release's signing and timestamp certificate chains and revocation status. TCP 80/443 access to the PKI/CRL/OCSP endpoints specified in the certificates may be needed. These download and trust connections do not carry printer telemetry.

Minimum operational rule. For a native installation updated manually, API TCP/443 and the selected printer protocols are sufficient, alongside name resolution and required response traffic. Approve the actual URL redirect chain for the specific release download. Only enable automatic updates with known destination hosts tied to that release; general internet access is not required.

The cloud providers' internal data flow does not require the customer's agent to connect directly to a database, Google Cloud or AWS. Website and installation traffic must be assessed separately from the persistent collection service.

Discovery and query protocols

Read operations and encryption

SNMP: the agent uses GET and limited table queries; it does not send SNMP SET. Provide read-only access on the printer. `snmp.min_version` is the lowest version tried: with `v1` (the default), configured `v3` is followed by `v2c` and then `v1`; with `v2c`, `v1` is skipped; with `v3`, only `v3` is tried and no community is sent. The `v1/v2c` community and traffic are unencrypted; the factory `public` value is used only when no community is configured.

Strict SNMPv3 environments: SNMPv3 only provides authentication and encryption with an appropriate `authPriv` configuration; MD5 or DES credentials load only with `snmp.allow_weak_v3_crypto: true` (see 09, F1).

IPP: only `Get-Printer-Attributes` is used; no print jobs are submitted. Direct IPP uses unencrypted HTTP by default. IPPS can use the operating system's certificate trust or a pre-approved SHA-256 certificate fingerprint for a specific printer. If IPPS validation fails, that IPPS route does not fall back to unencrypted IPP.

Scope, frequency and load

The address list received from the cloud takes precedence over the local `cids` setting, but only within two limits. It may contain only private, CGNAT (`100.64.0.0/10`) or IPv6 ULA addresses, or such ranges of at most 65,536 addresses; the server refuses public, loopback, link-local or cloud metadata addresses (such as `169.254.169.254`), and the agent skips and reports them. The local `remote_config.scan_ranges` setting, when bounded, narrows the cloud list to the part covered by the local `cids`; when `locked`, it excludes the cloud list (see 09). Without a cloud list, the local list applies, or failing that the ranges of local private-network interfaces. Every printer contact, including known printers, stays within this scope. The agent reports the entries of its local list that the cloud may also set, so the web app can show addresses outside the cloud list; only an authorised user can add them to the cloud list. Filtering VPN, tunnel and container interfaces is not a substitute for an IT-approved address list and firewall rules.

Operation / limit	Default and exceptions
Rediscovery	Every 30 minutes; also at startup, on scope changes and on operator request. In an active/standby group, the standby agent does not perform routine collection.
Meters / supplies / status / heartbeat	Meters and supplies: daily by default; configurable separately per network from hourly to every 90 days. Status: every 15 minutes; heartbeat: every 5 minutes. A new device, missed measurement, billing-cycle boundary or manual request may trigger additional reads.
Discovery budget	8192 addresses per cycle by default, with a 50 ms start interval (up to approximately 20 new addresses/s). These are configuration values, not performance guarantees for every release.
Concurrency / timeout	General default: 16 discovery workers; native Windows Setup configures 64. SNMP timeout: 2 s, 1 retry. An address may receive several protocol requests.

Restricted configuration without discovery

Merge the following fragment into the corresponding fields of a valid `agent.json`. Replace the example address with an approved printer IP; the cloud-configured addresses must also match.

```
{
  "cids": ["192.168.10.25"],
  "discovery": { "scanless": true },
  "ipp": { "disabled": true },
  "updates": { "mode": "managed" },
  "remote_config": { "scan_ranges": "locked" }
}
```

This retains SNMP-based operation and disables range expansion, mDNS, WSD and IPP. To retain IPP, set `disabled` to `false`; to disable only mDNS separately, use `ipp.mdns_disabled: true`. Restart the service after a local change. An IP list alone is not a substitute for `scanless: true`. With `locked`, the cloud cannot widen the scope.

Device identity: a unique serial number reported by the printer is required; an IP address, advertised name or UUID alone is insufficient. The code verifies the serial number again on every measurement. This reduces matching errors; on an unencrypted protocol it does not provide cryptographic device authentication.

Installation and runtime permissions

Installation requires local administrator privileges, but no PrinterMS user password. The service starts at system startup and the service manager may restart it after a failed exit.

Installation method	Runtime privileges	Impact and support scope
Native Windows	Collector: NT SERVICE\printerms-agent. Two separate maintenance services, started on demand: LocalSystem.	Primary installation method. Technical target platforms: x64 Windows 10/11 and Windows Server 2016 or later; install only on an OS release supported and kept up to date by the customer.
Linux / systemd	Dedicated printerms-agent user. Separate, short-lived root helper for maintenance.	x64/arm64 release; read-only system area, restricted writable state directory, empty capability set, and device and kernel restrictions.
macOS / launchd	The collector LaunchDaemon runs as the hidden, login-less _printerms account and can only read agent.json. Separate root LaunchDaemons for updates and remote decommissioning (E2).	x64/arm64 binary. Signing/notarisation and installation evidence needed for a managed Mac environment must be checked for each release; availability of a binary alone is not that evidence.
Docker	The release Dockerfile uses a non-root user (65532). State requires a writable volume.	IT replaces the container image. LAN routing and multicast reachability depend on host/bridge/Desktop networking mode and must be checked separately.

Services and default locations

Windows program / configuration	C:\Program Files\PrinterMS\Agent\printerms-agent.exe C:\Program Files\PrinterMS\Agent\agent.json
Windows state / installer	C:\ProgramData\PrinterMS\Agent C:\ProgramData\PrinterMS\Installer (installation, recovery and diagnostic files)
Windows services	printerms-agent printerms-agent-decommission-helper printerms-agent-update-helper
Linux	/usr/bin/printerms-agent /etc/printerms-agent/agent.json /var/lib/printerms-agent; protected /run and update-state areas are also used during maintenance.
macOS	/usr/local/bin/printerms-agent /Library/Application Support/PrinterMS/Agent (agent.json; state: Data) /Library/LaunchDaemons/com.printerms.agent.plist; separate update and decommission LaunchDaemons are also installed.
Docker	Persistent volume selected by data_dir / DCA_DATA_DIR. The actual path and restart policy are part of the container deployment specification.

Limits of host changes. The installer creates its own directories, services, ACLs and uninstall entry. Printer drivers and the printing path are unchanged. Optional `ipp-usb` is separate software with its own USB permissions.

Local data storage and retention

File / group	Content	Retention / deletion
agent.json device_token	Network and operational settings; SNMP credentials if configured; the agent's API token in a separate file.	For the installation's lifetime, until changed or the token is rotated. Delete on full removal.
outbox.db	bbolt database: JSON measurement packages awaiting delivery; a separate area holds permanently rejected or invalid packages.	A successfully accepted package is logically deleted. A rejected package remains for diagnostics/resubmission; there is no automatic time-based local expiry.
known_devices.json	Serial numbers, addresses, IPP routes, system descriptions, query mode, last successful observation and scheduled measurements.	The device registry is updated; a printer not verified for 7 days may be removed. On-disk state is updated during operation.
reset_baselines.json anchor_state agent_role	History used to detect meter resets, billing-cycle boundary and active/standby role.	Until overwritten by new state or fully removed; there is no common time limit for all files.
agent.log and its backups	Startup, connections, discovery, errors and operations. May include IP addresses, serial numbers, local paths and technical error text; the activation code only masked (see 09, E4).	Size-based rotation: 5 MiB active log, up to 3 earlier files. This is a size limit, not retention measured in days.
Definitions, updates and installer state	Signed definition/manifest, anti-rollback state, downloaded candidate, previous binary and transaction diagnostics.	According to the update/installation lifecycle and cleanup. A failed operation may leave recovery files.

Storage protection

Service-identity ACLs on Windows, and a dedicated user with restricted file permissions on Linux and macOS, protect the agent's data. On Windows the device token is DPAPI-protected in machine scope (E1); the outbox, and the token on other systems, have **no separate application-level encryption**. Host IT policies provide any required disk encryption, backups and protection of administrator access.

Logical record deletion is not secure disk erasure: bbolt reuses freed pages and does not shrink the file after every acceptance. Administrator access can also reach local data. Do not use the token file or the full data directory to clone another agent.

Internet outages and a full outbox

Measurements already written to disk survive a restart. The agent retries transient connection failures after a delay; it does not delete a package until the API accepts it. Server acknowledgement confirms acceptance; subsequent processing is a separate step.

Total local capacity is **5000 packages or 512 MiB of logical package content**, including rejected packages. Within this, 200 packages and 32 MiB are reserved for error reports. Total disk usage may be higher because of the physical database file and other files.

Limits of outage protection. Existing evidence is retained when the outbox is full, but storage of new snapshots may be restricted. Short-lived errors may be missed; a stopped or sleeping host does not measure. The outbox does not provide unlimited outage tolerance or a general guarantee against data loss.

Retention for the systemd journal, separate macOS stderr log, Windows installer logs and container logs must be managed separately; the agent.log rotation limit does not apply to them.

Cloud data processing and access

The agent sends data to its assigned organisation and network through the `api.printerms.com` API. The API determines the agent’s permissions from its device token; telemetry cannot freely specify another organisation ID. Once approved, a code-activated agent is trusted from its first measurement; an authorised organisation user approves it with knowledge of the host’s details. Viewing is restricted by organisation permissions and database row-level security (RLS).

Service component	Role in the DCA data flow	Documented location
Google Cloud / Cloud Run	API and background processing; handling technical requests.	Belgium, europe-west1.
Google Cloud	Cloud database and file storage, with daily backups.	Belgium, europe-west1; backups within the EU.
Amazon Web Services	Authentication of PrinterMS users (Amazon Cognito).	Ireland, eu-west-1.
Cloudflare	Website, download page and DNS/CDN delivery; web request metadata.	Global edge network; not the agent’s local measurement database.
Email and support	Enabled notifications and separately submitted support requests.	As listed in the subprocessor register. Support correspondence is not exclusively EU-region based.

Alongside EU-region primary storage, incidental international support or provider access is possible. Providers and transfer conditions: printerms.com/en/subprocessors.

Access and responsibility

Users in the printer operator’s organisation with appropriate viewing permissions can access their own data; configuration, token management and removal require separate edit/delete permissions. PrinterMS service operations may require privileged system access; RLS is not encryption against the database operator.

The printer operator contracting directly with the customer determines the purposes of collection and authorised recipients; the data processing agreement sets out PrinterMS’s role as processor. These terms also apply to technical data that may relate to a person. Confidentiality, incident notification and audit terms are linked at the end of this document.

Retention and deletion

Cloud data category	Documented behaviour
Completed raw submission jobs	Default cleanup: 7 days for meter and supply data, 30 days for other submissions. In-progress/retryable work is not a completed record of this kind.
Detailed DCA technical history	Supplies: 90 days by default; status: 30 days; resolved alerts: 180 days from resolution. Active errors and the current device snapshot are not deleted under these time limits.
Meter readings and business device records	Part of the organisation’s dataset. Removing an agent or deactivating a device does not automatically delete historical data.
Device-support walk and photo	Deleted after 12 months, while the request outcome remains; visible only to PrinterMS staff. Deleted when the organisation is deleted.
Trial-rule results	Deleted after 30 days; not displayed or billed.
Account closure / data deletion	Export, 30-day return/grace period and permanent deletion under the data processing agreement, with statutory retention exceptions.

Technical history windows are server-side defaults; production configuration and cleanup-job operation determine actual execution. Where contractually required, retention for the specific data category must be recorded in writing.

Backup and recovery. Under section 9.3 of the data processing agreement, the database is backed up automatically each day with encryption within the EU, with seven-day point-in-time recovery (PITR) and weekly full exports; backups are retained for up to 42 days. The local outbox is not a cloud backup. Exports can be used for your own archive; a separate RPO/RTO commitment is only assured by explicit agreement.

Activation, remote operations and updates

Code activation and approval

After installation, the agent displays an ABCD-EFGH activation code, valid for 60 minutes and automatically renewed for up to 7 days. In **Networks** → **New agent**, a PrinterMS user with `agents.create` permission can reject the request or assign the host to a new or existing network, based on host name, version, source IP and detected subnets. Incorrect codes are limited to 10 attempts per user per 15 minutes and 30 per organisation per hour.

The short code does not authorise token collection: only the holder of the separate 256-bit private device code can collect the device token. The token is machine API access, not a user session; it can be revoked and rotated. The agent shows where it has connected; with notifications enabled, approval triggers a security email.

Install link: a user with `agents.create` permission can create a revocable link or email campaign bound to one organisation, client and network, valid for up to 30 days and 1-10 installations. Of the 128-bit token the server stores a SHA-256 hash and an encrypted copy. With a valid token the request is approved automatically into the bound network; the device token is still collected only with the 256-bit device code, Setup does not store the token, and a rejected token falls back to code activation. Every use sends an email and is logged; Windows Setup adds a read-only agent status shortcut.

Operations communicated by the cloud

Operation	Effect and limit
Configuration	Address ranges, reading schedule, collection categories and active/standby role; enabling WSD and the local USB connector. The cloud address list applies only within rules B1 and B2 of chapter 09.
Immediate query	Discovery, meter, supply or status query. The handler accepts only predefined command names; there is no arbitrary shell command or uploaded script.
Device-support walk	Only on an informed, explicit request by a user with <code>agents.edit</code> permission, for an already discovered printer. If the Owner separately opts into the device-support programme (off by default), PrinterMS may schedule walks overnight (20:00-06:00) for printers seen in the last 48 hours: up to ten per run, and at most one per model and firmware version across all customers. The agent reads the built-in branches once, at a reduced rate, outside printing, for up to 5 minutes. It does not read the print log, address book, users or network tables. Serial numbers, IP and MAC addresses, email addresses, host names, phone numbers and long text are redacted locally (09, F6), and the server rechecks them.
Local restrictions	<code>scanless</code> , disabling IPP entirely, local managed update mode and <code>support.device_walks: disabled</code> (or <code>DCA_DEVICE_WALKS=disabled</code>) restrict the corresponding operations. The walk contents can be previewed locally: <code>printers-agent --walk-device <printer-IP> --walk-out <new-file></code> writes the same data to a local file and sends nothing.
Token and decommissioning	Token rotation, revocation and remote self-removal. Decommissioning may use the narrowly scoped privileged helper created during installation.
Query definition	A separately signed, restricted data format may change supported reading rules. It cannot provide executable code, a general script or a new network destination.

The agent receives instructions in responses to its own HTTPS requests. They do not run immediately on an offline host. Local logs contain operational events; changes made through the cloud are recorded in the organisation's log (09, B3).

Software updates and release trust

Update mode: the organisation or network setting in PrinterMS decides; the `updates.mode: automatic` of a new installation follows it. IT can restrict it on the command line: with `updates.mode: approval` a release installs only on request, with `managed` never. Every self-update needs at least two signatures and a 24-hour wait (09, A1, A3, A4).

Native Windows Setup: a separate native process running as LocalSystem performs the replacement, checking the manifest signature, SHA-256 hash and pinned PrinterMS Authenticode identity. After replacement it waits for an authenticated status report and restores the previous binary on failure; it does not use PowerShell.

Other native methods: verify and roll back the same way; their widespread production use is not evidenced.

Docker: a container does not rewrite its own image; IT deploys the new image. `managed` mode only prevents executable replacement.

Endpoint protection. Signatures provide verifiable origin; they do not replace EDR policies. Tie exceptions to a verified publisher or exact hash and a narrow path. VirusTotal receives only public release files, never customer data.

Threats and built-in safeguards

The table below lists the failure and misuse scenarios that the agent and the PrinterMS cloud protect against in code, together with the limits of each safeguard. The row identifiers are stable, so an approval record can refer to them. The third column shows what IT can configure locally and what risk remains.

Threat	Built-in safeguard	Local setting and residual risk
B1 · Cloud address list A faulty or unauthorised cloud setting directs SNMP queries, including the unencrypted v1/v2c community, to an internet address, the host itself or a cloud metadata service.	The cloud can set only private (10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16), CGNAT (100.64.0.0/10) or IPv6 ULA (fc00::/7) addresses or ranges, each covering at most 65,536 addresses: a prefix of at least /16 for IPv4 and at least /112 for IPv6. It cannot set a public address, loopback (127.0.0.0/8, ::1), link-local (169.254.0.0/16, fe80::/10), a cloud metadata address (169.254.169.254, 100.100.100.200, fd00:ec2::254) or a range containing one, multicast, broadcast or 0.0.0.0/8. The server refuses a new cloud entry and drops disallowed ranges from installer suggestions. The agent applies the same rule to the list it receives: it skips and reports a disallowed entry, and if nothing remains it uses its local allowance.	Public ranges can be set only locally, in the agent.json cidrs field; this is an explicit IT decision. A previously stored cloud entry that is no longer allowed is kept and does not block other changes; the network page flags it, and it can be removed in one step. An older agent that does not know the rule still uses such an entry, so it should be removed.
B2 · Local cap The cloud list makes the agent scan a wider network than approved locally, switches WSD or the local USB connector on from the cloud, or keeps querying a previously known printer that is now outside the scope.	In agent.json, the remote_config.scan_ranges field (or the DCA_REMOTE_SCAN_RANGES environment variable) caps the cloud list. open (default): the cloud list that passes rule B1 applies. bounded: only cloud entries covered by the local cidrs, or by the host's own networks if that list is empty. locked: no cloud range at all, only the local allowance. Every printer contact (discovery, meter, supply and status queries, device-support walks) passes one scope check, including contacts with printers already known: a printer outside the scope receives no request and is removed from the registry. The agent reports the mode, the skipped ranges and the number of printers outside the scope; the web app shows them on the network page.	The default is open, so IT has to set the cap; a change takes effect after the service restarts. In bounded and locked mode the cloud cannot switch WSD or the local USB connector on, only off. On a network that shares its workload between agents, a printer is read only if both the server-assigned work scope and the local cap cover it. The scope does not apply to the host-local IPP-over-USB connector.

Threat	Built-in safeguard	Local setting and residual risk
C1 · Meter jump A misreading or a tampered printer reports an implausibly large meter increase in a single reading, in many small steps over one day, or after a reading that left the meter out, which would end up as a charge on the invoice.	On the Platform plan the server measures every agent reading against the device's previous reading, and each billed meter (black and white, colour, scan) against that meter's last known value, the latest agent reading of an earlier day that carries it, searched back to the reading billing compares it with (the last billed, delivery or acquisition reading). If the printer is billed from the agent, each meter is also measured against that billing reading, which also replaces a missing or older last known value, so a jump is held even on the printer's first agent day or on the date of that reading, and when a delivery or acquisition reading is entered later with an earlier date. If a meter grows by more than 20,000 pages per elapsed day against the previous reading (counting at least one day), or per calendar day against its last known value or the billing reading, the server holds the device's meters (<code>counter_jump</code>): no reading is created and nothing is billed until a user with <code>readings.financial.manage</code> and <code>printers.edit</code> decides on the printer page, after confirmation. <i>New baseline</i> : the next reading becomes a free starting point. <i>Bill the measured value</i> : the next reading is measured against the previous reading only and billed against the last billed reading, the jump included; later readings of that day are not stored, and the next day is measured against it. After either release, readings are measured again against a billing reading entered later, even with an earlier date, and the release ends when the device is linked to another printer or unlinked. A break caused by a changed meter interpretation (<code>counter_rule</code>) is held on every plan and can be released only with a new baseline.	The limit is not configurable, so a real, exceptionally heavy printing day is held too. A printer billed from another source, such as e-mailed meter reports, is not measured against the billing reading, only against its own agent readings, as that source may count differently; an agent reading billing a month that source missed is billed only within the same limit of the billing reading. Agent readings stored while the printer was billed from another source are not re-checked against the billing reading when the printer is later switched to billing from the agent. After a release, readings are not measured against a billing reading that already existed when the first reading after the decision was stored and is not dated later than that reading. A decrease without an interpretation change is not held; billing treats it as a new starting point. In PrinterMS Monitor agent readings are not billed, so there is no jump hold. PrinterMS can switch the safeguard off temporarily during an incident; existing holds stay in place.
D1 · Device takeover A standby or foreign node, or an agent of another network, reports the same printer and takes over or duplicates the billed series with its own meters.	Within a network, a device stays bound to the node that reported it. Another node takes it over only if it is set to active or holds the active slot and its observation is not older, or if the current node is no longer active (for example revoked). A report from any other node updates identity, supply and status data, but its meters are neither stored nor billed. On the Platform plan, if a device of another network already counts the printer, the new device is not linked to it and the app names the counting network; linking it (<code>agents.edit</code> and <code>printers.edit</code>) unlinks the other network's device in the same step.	Assigning the active role (<code>agents.edit</code>) can still move the counting. With overlapping address ranges, the second network's devices appear unlinked. Conflict handling in PrinterMS Monitor is unchanged. PrinterMS can switch the safeguard off temporarily during an incident.
E4 · Activation code in logs A third party reads a still valid activation code from a log attached to a support ticket, forwarded to a log collector or readable by others, and approves the host in their own PrinterMS account.	Agent and Windows Setup log records (for example <code>agent.log</code>) contain the code only masked (<code>AB*-****</code>): the first two characters remain, the rest of the code and its length do not. Only the output meant for the operator shows the full code: the console message, the <code>--activation-code</code> command, the local activation status file and the Setup window. A code is valid for 60 minutes; after approval the agent shows which organisation it has connected to.	The console message goes to the service's standard output, so on Linux the systemd journal and in Docker the container output show the full code; restrict access to them. Logs of earlier agent versions may contain the full code.
F1 · SNMP fallback An agent configured for v3 falls back to v2c or v1 and sends the community unencrypted, possibly the factory <code>public</code> value, or uses weak SNMPv3 credentials based on MD5 or DES.	<code>snmp.min_version</code> is the lowest version tried. <code>v3</code> : v3 only, no community reaches the network. <code>v2c</code> : no v1. A known printer is not queried over a previously stored route below the minimum either. <code>public</code> is used only when no community is configured and v1 or v2c is permitted. SNMPv3 credentials using MD5 or DES fail configuration loading unless <code>snmp.allow_weak_v3_crypto</code> : <code>true</code> is set. The agent reports both settings to the cloud.	The default is <code>v1</code> , so IT has to set the stricter value. With <code>v3</code> and no <code>snmp.v3</code> username the agent reads no printer and only sends its heartbeat. <code>v1/v2c</code> traffic stays unencrypted; older versions should also be restricted on the printer and network.

Threat	Built-in safeguard	Local setting and residual risk
F6 · Free text in walks A name given to the printer, or a name, department or note typed into a manufacturer field, reaches PrinterMS through a device-support walk.	The agent drops the printer's own name (<code>prtGeneralPrinterName</code>) from the walk, next to the system contact, location and operator fields. In manufacturer branches it replaces every text longer than 16 bytes locally with <code><redacted text len=N></code> , except numbers, dotted version numbers and hex digits; serial number fields and the numeric values kept as hex are exempt. A kept value still passes the address, serial number and contact filters. The server rechecks the upload.	Text in the standard branches and manufacturer text of up to 16 bytes (such as supply names) is kept. Earlier agent versions may also upload the printer name. The walk can be previewed (<code>--walk-device</code>) and forbidden locally (<code>support.device_walks: disabled</code>).
A7 · Decommissioning and deletion A user without the authority decommissions or deletes a network or node from the cloud and so stops counter collection for its printers.	Decommissioning and forced deletion of a network or node require the <code>agents.delete</code> permission; without it the server refuses with 403 and the interface does not offer the action. The default Technician template does not include it, and only the Owner can grant it. Every decommission request and deletion is written to the organisation's log (B3).	The Owner can grant the permission to another member through a template; that is deliberate delegation. With local administrator rights the service can be stopped or removed on the host without the cloud.
B3 · Untraceable change After a network setting, command, activation, revocation or counter hold release, nobody can tell who made the change and when.	The cloud writes every agent network change (settings and update policy, commands, activation and refusal, revocation, decommissioning and deletion, token rotation, device assignment, counter holds and their release) to the organisation's log in the same transaction as the change: time, actor (user, agent or system), network and node. An entry cannot be changed, and cannot be deleted while the organisation exists; entries of deleted networks and nodes remain. No token, code, community or password is written to it. Users with <code>agents.view</code> see it in the Log section of the Networks page and of the network page.	The log records changes made through the cloud; it does not record edits to the local <code>agent.json</code> or actions on the host, for which the host's own logs serve. Entries are deleted with the organisation.
C3 · Activation request flood An attacker sends anonymous activation requests in bulk to use up the allowance of waiting requests, so that genuine installations cannot activate.	The anonymous activation endpoints accept 10 requests per minute per caller address (per /64 for IPv6). They are counted in a shared database; if it is unavailable, the server refuses the request (429, retry after one minute) instead of letting it through. At most 20 requests may wait at once from one source network (IPv4 /24, IPv6 /48), and 5000 across the service. Above 2500 waiting requests a new code is valid for 15 minutes instead of 60. The agent retries by itself after a refusal or expiry, and no request gains access without approval.	From a shared exit address (NAT, CGNAT) or one /24 network, at most 20 machines can wait for approval at once; the others get a code later. While the limit counter is unavailable, no new activation starts. A flood from many source networks can delay new activations across the service; agents already activated are not affected.
C4 · Replaced token An old device token left on a cloned, restored or stolen machine still reaches PrinterMS after a token rotation.	On rotation the old token stops working when the agent first connects with the new one, which proves it has saved it; until then the old one works, so a rotation cannot lock the agent out. Every later use of the replaced token is refused with 401. If more than 10 minutes have passed since the rotation, a <code>node.token_stale_use</code> event is written to the organisation's log (at most one per node per hour) and the server logs a warning. After revocation or decommissioning no token of the node is valid.	Rotation and revocation are started by a user with <code>agents.edit</code> . The server cannot tell a copy of the current token from the original; on suspicion, rotate or revoke it. No separate email is sent for the event.
A6 · Install command <code>agent.printerms.com</code> , which serves the short install command, or its deployment pipeline writes a foreign hash, a weaker Windows check or another signer into the bootstrap script, and the host runs foreign software as administrator.	On Windows the production bootstrap script accepts only the PrinterMS Authenticode signature with a fixed signer identity; SHA-256-only checking is limited to the test environment, and with any other setting the production server serves no script. The production server accepts supplied hashes only from the <code>agent-release.yml</code> release workflow of that release, after its test-environment qualification and protected approval; a manual update derives them from the latest release. Every hash is compared with the downloaded files of the release before going live.	The bootstrap script and its hashes come from the same server, so the check independent of it is the release's signed <code>checksums.txt:cosign verify-blob</code> with <code>checksums.txt.sig</code> and <code>checksums.txt.pem</code> downloaded from the GitHub release, against the release-bound identity of the <code>agent-release.yml</code> workflow. The check runs on any workstation with internet access; the collector host needs no Sigstore access.

Threat	Built-in safeguard	Local setting and residual risk
A1 · Update mode A new release installs on every host at once, without a decision by IT.	The organisation or network setting in PrinterMS decides, and the local automatic mode of a new installation follows it: set to automatic, a verified release installs without a request after the 24-hour wait (A3); set to managed, only when a member with <code>agents.edit</code> requests it. The stricter of the local and the cloud mode applies.	The decision is in the cloud; A3 and A4 protect against a compromised account. Only IT sets an independent local limit, on the command line: <code>--update-mode</code> or <code>DCA_UPDATE_MODE</code> (approval, managed); the app shows when it is stricter. In approval mode the computer's administrator can also approve (tray icon, UAC, logged).
A3 · Immediate release A compromised release pipeline publishes a signed but faulty or malicious release that reaches hosts before it is detected.	The agent downloads and installs a release 24 hours after the later of the signed issue time and its first verification on the host, so backdating or a restart does not shorten the wait. Until then it reports <code>soak_waiting</code> and the end of the wait, shown in the app as "Safety wait". In that time PrinterMS can stop the release.	The wait also applies to urgent fixes; a faulty new version is covered by health-check-based rollback. IT replaces the Docker image, so there is no wait there.
A4 · Signing key A single stolen or misused signing key approves executable code for the hosts.	The update manifest is authenticated by the signatures of at least two independent keys: the agent, the trust root tool and the release check refuse a threshold below 2, and with such a root the agent does not update. The privileged helper re-verifies the signatures and the file hash before installation.	The threshold means two independent decisions only if different people or systems hold the keys. Both keys are currently in one cloud project under one owner identity; separating them is a planned PrinterMS task.
E1 · Device token at rest Someone uses a device token copied from a disk image, a backup or the host to send data or receive instructions as the agent.	On Windows the token is stored DPAPI-protected in machine scope (<code>dpapi:v1:</code>): a fresh Setup writes it that way, an existing installation converts it after the current Setup has run, at its first authenticated heartbeat outside an update transaction. On Linux, macOS and Docker the token is a 0600 file read only by the service account and root. The agent reports the storage form (<code>token_storage</code>); rotation or revocation invalidates a copied token (C4).	DPAPI protects a copy taken off the host, not against a local administrator or code running as SYSTEM. A host installed with <code>install.ps1</code> or only self-updated from an older Setup keeps the plaintext file until the current Setup runs. Disk encryption is the task of IT.
E2 · Root collector on macOS Through a flaw in the collector, which parses printer responses, an attacker runs code with root privileges on the Mac.	<code>install.sh</code> runs the macOS collector as the hidden, login-less <code>_printerms</code> account (system ID below 500, shell <code>/usr/bin/false</code>). It can only read the root-owned <code>agent.json</code> and write only its own <code>Data</code> directory. Updates and remote decommissioning are done by separate root LaunchDaemons: they accept only files of root or this account, take the paths they delete from root-owned files, and install an update only from a copy verified in their own root directory.	An existing Mac switches when IT runs <code>install.sh</code> again (the macOS agent does not update itself); until then the collector runs as root. The root helpers remain privileged code, and a local administrator can change the agent. On Windows, Linux and Docker the collector already runs without administrator rights (05).
G1 · Cloud connection A modified setting, a redirect or an intercepting network device sends the device token and the measurement data to a foreign server.	<code>server_url</code> can only be an address in the <code>printerms.com</code> domain, otherwise the configuration does not load; only a local setting can allow another host, <code>allow_custom_server_url</code> or the <code>DCA_ALLOW_CUSTOM_SERVER_URL=1</code> variable, never the cloud. The agent never follows an API redirect, uses at least TLS 1.2 and validates the certificate against the OS trust store. A configured <code>proxy_url</code> (or <code>DCA_PROXY_URL</code>) applies to every cloud request and takes precedence over the <code>HTTPS_PROXY</code> , <code>HTTP_PROXY</code> and <code>NO_PROXY</code> variables. In <code>strict_tls</code> mode only <code>proxy_url</code> counts, and the certificate chain of the <code>server_url</code> host must contain a <code>server_spki_sha256</code> key.	By default there is no key pinning, and without <code>proxy_url</code> the proxy environment variables apply. With <code>strict_tls</code> , corporate TLS inspection works only if the key of the inspecting certificate authority is pinned, and the pins must be updated when keys change. Pinning does not cover the update file host; signatures protect those files (A4). Older agent versions do not know these rules.
G2 · Security contact Someone who finds a vulnerability in the agent or the PrinterMS cloud does not know where to report it, so the flaw becomes public before PrinterMS can fix it, or never reaches PrinterMS.	The <code>https://www.printerms.com/.well-known/security.txt</code> file (RFC 9116) names the reporting address (<code>support@printerms.com</code>) and Hungarian and English as languages, and points to the "Reporting a vulnerability" section of the Security page . The rule: the reporter writes privately by email, PrinterMS acknowledges within 5 working days, and the details stay private until the fix, for no longer than 90 days.	Reports arrive in the support mailbox; there is no separate security address and no paid bug bounty programme. PrinterMS renews the file's expiry date every year.

Firewall rules remain an independent, customer-side limit for every row.

Operational verification and removal

Record on approval

Host and responsible person; installed version and release SHA-256; approved printer IPs/CIDRs; the local cap mode for the cloud address list; permitted protocols; users authorised to make cloud changes; reading schedule; update mode; protection of the local data directory; cloud retention; and the person responsible for removal.

Verify connectivity, proxy and operation

- 1. Release:** compare the installer’s signature and SHA-256 hash with the approved release. Checking the publisher signature and checking the hash are separate steps.
- 2. API connectivity:** `curl.exe -fsS https://api.printerms.com/healthz` on Windows, `curl` on other systems. This verifies DNS/TLS/connectivity for the test process; a successful authenticated heartbeat proves it for the service.
- 3. Proxy:** preferably set `proxy_url` in `agent.json` (or `DCA_PROXY_URL` in the service’s environment); it takes precedence. Without it the `HTTPS_PROXY`, `HTTP_PROXY` and `NO_PROXY` variables of the service’s own environment apply; with `strict_tls` the agent ignores them. Cloud TLS validation uses the OS trust store, with key pinning only under `strict_tls` (09, G1); corporate TLS inspection requires a root certificate trusted by the service. Printer SNMP/IPP does not pass through the web proxy.
- 4. Printer:** compare an approved device’s serial number, meter and status with the data shown in PrinterMS. Check actual destinations and frequency against firewall/EDR logs.
- 5. Continuity:** verify restart, outbox draining and outage alerts. Measure CPU, RAM, disk and traffic with the actual fleet; there is no universal load guarantee.

Immediate local stop without internet access

The following administrator operations stop the collector and disable its normal automatic restart. They do not delete data or revoke the cloud token.

Windows (administrator terminal)	<code>sc.exe config printerms-agent start= disabled</code> <code>sc.exe stop printerms-agent</code> <code>sc.exe query printerms-agent - expected state: STOPPED.</code>
Linux	<code>sudo systemctl disable --now printerms-agent</code>
macOS	<code>sudo launchctl disable system/com.printerms.agent</code> <code>sudo launchctl bootout system/com.printerms.agent</code>
Docker	<code>docker update --restart=no printerms-agent</code> <code>docker stop printerms-agent</code> ; with an orchestrator, also change its desired replica count/restart configuration.

Running maintenance and external deployment controllers need separate handling. Network isolation is also an option during an incident. Reconcile local and cloud configuration before re-enabling.

Full removal and data deletion

Windows: Settings → Apps → PrinterMS Agent → Uninstall. The native uninstaller requires server-side token revocation; if this cannot be confirmed, it preserves/restores the installation for a retry. For an immediate offline block, use the stop procedure above. Do not manually delete the program directory before running the uninstaller.

Linux/macOS/Docker: after revoking the token in the cloud, remove the service/package or container, configuration and persistent data directory/volume. On Linux, ordinary package removal may leave data; perform full cleanup according to the package and installation method. After local removal, historical cloud data is subject to a separate deletion procedure.

Documents and contact

[Data processing agreement](#) · [Subprocessor register](#) · [Privacy notice](#) · [DCA terms](#) · [Provider details](#)
Support and security incidents: support@printerms.com · Data protection: privacy@printerms.com