

Private Cloud Option for Agentic Dental Operations System.

Infrastructure for AI Agent Operations in a Dental Practice

Document type: Infrastructure specification and implementation guide

Version: 2.0 — September 3, 2026 (supersedes v1.0 shared-hosting draft)

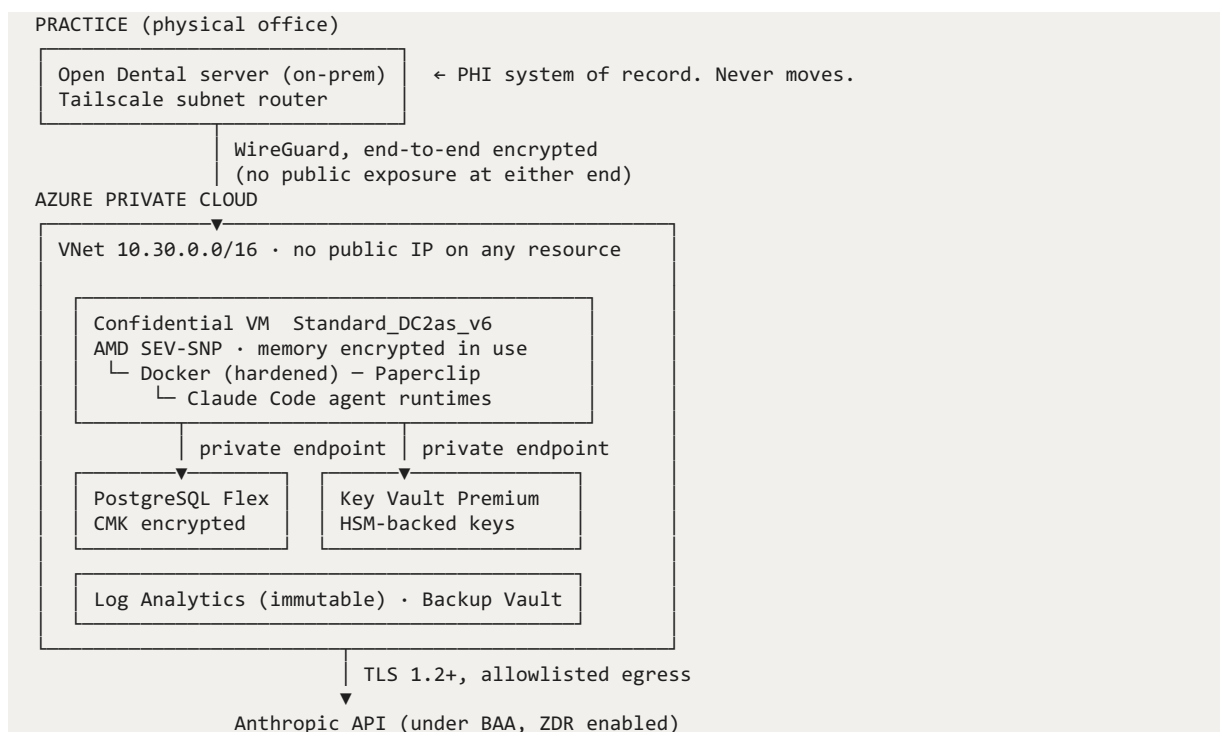
Platform: Microsoft Azure confidential computing

Applies to: Paperclip.ing agent control plane with Open Dental integration

IT Implementation Guide

Written to be executed step by step by an engineer or an AI agent with Azure CLI access. Every phase ends with a verification command. Do not proceed past a failed verification.

2.0 Architecture



Data classification. PHI in this environment is limited to: patient name, one contact method, proposed treatment description and value, and last visit date. Clinical notes, radiographs, insurance identifiers, Social Security numbers, guarantor financials, and payment instruments are **out of scope and must never be queried**. This is the minimum necessary determination and it is enforced at the Open Dental query layer, not left to agent judgment.

2.1 Prerequisites

- Azure subscription under a Microsoft Customer Agreement or Enterprise Agreement (either incorporates the Product Terms and the Data Protection Addendum, which is what carries the HIPAA BAA).
- Azure CLI 2.60+ authenticated as Owner on the target subscription.
- Entra ID tenant with MFA enforceable.
- Anthropic organization: 1P API with HIPAA enabled and ZDR, or Claude Enterprise with HIPAA activated. Start this early; it involves Anthropic Sales and is the long pole.
- Open Dental API credentials (Developer Key + Customer Key) obtained from the practice.
- A Tailscale organization with an admin account.

Variables

Set once. Every subsequent command references these.

```
export LOC="eastus2" # verify DCasv6 availability first (2.2)
export RG="rg-pclip-prod"
export VNET="vnet-pclip"
export SUBNET_APP="snet-app"
export SUBNET_PE="snet-privatelink"
export NSG="nsg-pclip-app"
export KV="kv-pclip-$RANDOM" # must be globally unique
export DES="des-pclip-cvm"
export VM="vm-pclip-01"
export VMSIZE="Standard_DC2as_v6"
export PGSRV="pg-pclip-$RANDOM"
export LAW="law-pclip"
export BVAULT="bv-pclip"
export ADMIN="pclipadmin"
export TAG="env=prod purpose=paperclip-agent-controlplane data=phi owner=msp"
```

2.2 Phase 1 — Subscription, region, quota

```
az account set --subscription "<SUBSCRIPTION_ID>"

# Confirm the confidential SKU exists in the chosen region.
az vm list-skus --location $LOC --size Standard_DC2as_v6 \
  --resource-type virtualMachines --output table

# Confirm quota headroom for the DCasv6 family.
az vm list-usage --location $LOC --output table | grep -i "DCAS"
```

If the SKU is absent, try eastus, westus3, or southcentralus. Do not silently substitute a non-confidential size — that is a security architecture change requiring a documented decision.

Region rule: United States only, and record the paired region for backup.

```
az group create -n $RG -l $LOC --tags $TAG
```

Verify: `az group show -n $RG --query "properties.provisioningState"` returns Succeeded.

2.3 Phase 2 — Identity and access

```
# Register the confidential computing orchestrator service principal in the tenant.
# Required before a confidential disk encryption set can be created.
az ad sp create --id "bf7b6499-ff71-4aa2-97a4-f372087be7f0" \
  --only-show-errors || echo "Confidential VM Orchestrator SP already present"
```

In the Entra portal, establish:

- A security group pclip-admins containing only named MSP engineers. No shared accounts.
- Conditional Access requiring MFA for every member of that group on every Azure sign-in.
- Just-in-time elevation via PIM if the tenant is licensed for it; otherwise, standing RBAC with quarterly review.
- RBAC scoped at the resource group, never the subscription. Contributor on \$RG for pclip-admins; Key Vault Crypto Officer granted separately and to fewer people.

Verify: attempt a sign-in as a group member without MFA. It must be blocked.

2.4 Phase 3 — Network

No resource in this build receives a public IP address.

```
az network vnet create -g $RG -n $VNET -l $LOC \
  --address-prefix 10.30.0.0/16 \
  --subnet-name $SUBNET_APP --subnet-prefix 10.30.1.0/24 --tags $TAG

az network vnet subnet create -g $RG --vnet-name $VNET \
  -n $SUBNET_PE --address-prefix 10.30.2.0/24 \
  --disable-private-endpoint-network-policies true

az network nsg create -g $RG -n $NSG -l $LOC --tags $TAG

# Explicit deny-all inbound at priority 4096. Nothing above it permits inbound.
az network nsg rule create -g $RG --nsg-name $NSG -n deny-all-inbound \
  --priority 4096 --direction Inbound --access Deny \
  --protocol '*' --source-address-prefixes '*' --destination-port-ranges '*'

# Egress: allow only what the workload needs. Adjust the FQDN set after 2.11.
az network nsg rule create -g $RG --nsg-name $NSG -n allow-https-out \
  --priority 100 --direction Outbound --access Allow \
  --protocol Tcp --destination-port-ranges 443

az network nsg rule create -g $RG --nsg-name $NSG -n allow-tailscale-udp \
  --priority 110 --direction Outbound --access Allow \
  --protocol Udp --destination-port-ranges 41641 3478

az network nsg rule create -g $RG --nsg-name $NSG -n deny-all-outbound \
  --priority 4095 --direction Outbound --access Deny \
  --protocol '*' --destination-port-ranges '*'

az network vnet subnet update -g $RG --vnet-name $VNET -n $SUBNET_APP \
  --network-security-group $NSG
```

Verify: `az network nsg rule list -g $RG --nsg-name $NSG -o table` shows deny rules at 4095/4096 and nothing permitting inbound from Internet.

Administrative access arrives over Tailscale (2.10), not SSH from the internet, not Bastion with a public IP.

2.5 Phase 4 — Key Vault and confidential disk encryption set

Confidential VM disk encryption requires **Key Vault Premium** (HSM-backed). Standard SKU will fail.

```
az keyvault create -g $RG -n $KV -l $LOC --sku Premium \
  --enable-purge-protection true --retention-days 90 \
  --enable-rbac-authorization false --tags $TAG
```

```
# Exportable RSA-HSM key with the confidential VM release policy.
az keyvault key create --vault-name $KV -n cvm-osdisk-key \
  --key RSA-HSM --size 3072 --exportable true \
  --policy "@cvm-release-policy.json"
```

Where `cvm-release-policy.json` is Microsoft's standard confidential VM key release policy (obtain the current version from the Azure confidential computing documentation — do not author this by hand).

```
KID=$(az keyvault key show --vault-name $KV -n cvm-osdisk-key --query key.kid -o tsv)

az disk-encryption-set create -g $RG -n $DES -l $LOC \
  --key-url "$KID" --source-vault $KV \
  --encryption-type ConfidentialVmEncryptedWithCustomerKey

DES_ID=$(az disk-encryption-set show -g $RG -n $DES --query id -o tsv)
DES_PRINCIPAL=$(az disk-encryption-set show -g $RG -n $DES --query identity.principalId -o tsv)

az keyvault set-policy -n $KV -g $RG --object-id "$DES_PRINCIPAL" \
  --key-permissions wrapkey unwrapkey get
```

Then restrict the vault to the private network:

```
az keyvault update -g $RG -n $KV --default-action Deny --bypass AzureServices
az network private-endpoint create -g $RG -n pe-kv --vnet-name $VNET \
  --subnet $SUBNET_PE --private-connection-resource-id \
  $(az keyvault show -n $KV -g $RG --query id -o tsv) \
  --group-id vault --connection-name pe-kv-conn
```

Verify: `az disk-encryption-set show -g $RG -n $DES --query "encryptionType"` returns `ConfidentialVmEncryptedWithCustomerKey`.

Record for the safeguards summary: key name, vault name, HSM backing, purge protection window.

2.6 Phase 5 — Deploy the confidential VM

Enumerate the current confidential Ubuntu images rather than assuming a version string:

```
az vm image list --publisher Canonical --all --output table | grep -i "cvm"
```

Select the newest LTS confidential image. `Canonical:0001-com-ubuntu-confidential-vm-jammy:22_04-lts-cvm:latest` is known-good; use a newer noble image if listed.

```
az vm create -g $RG -n $VM -l $LOC \
  --size $VMSIZE \
  --image "Canonical:0001-com-ubuntu-confidential-vm-jammy:22_04-lts-cvm:latest" \
  --admin-username $ADMIN \
  --generate-ssh-keys \
  --security-type ConfidentialVM \
  --os-disk-security-encryption-type DiskWithVMGuestState \
  --os-disk-secure-vm-disk-encryption-set "$DES_ID" \
  --enable-secure-boot true \
  --enable-vtpm true \
  --public-ip-address "" \
  --nsg "" \
  --vnet-name $VNET --subnet $SUBNET_APP \
  --os-disk-size-gb 64 \
  --tags $TAG
```

`DiskWithVMGuestState` encrypts both disk contents and guest state (vTPM state, UEFI variables) under your key. `--public-ip-address ""` is deliberate and must not be removed for convenience during troubleshooting.

Add an encrypted data disk for Paperclip state:

```
az disk create -g $RG -n disk-pclip-data -l $LOC \
--size-gb 128 --sku PremiumV2_LRS \
--security-type ConfidentialVM_DiskEncryptedWithCustomerKey \
--secure-vm-disk-encryption-set "$DES_ID" --hyper-v-generation V2 --tags $TAG

az vm disk attach -g $RG --vm-name $VM --name disk-pclip-data
```

Verify (from inside the VM, once Tailscale access exists):

```
sudo dmesg | grep -i sev          # expect SEV-SNP active
sudo tpm2_pcrread sha256:0,1,7    # vTPM responding
lsblk                             # data disk present
```

Known operational constraint: DCasv6 does not support live migration or memory-preserving updates. Azure host maintenance reboots this VM. Configure a maintenance window outside practice hours and confirm the stack returns cleanly.

```
az maintenance configuration create -g $RG -n mc-pclip -l $LOC \
--maintenance-scope InGuestPatch --start-date-time "2026-09-07 02:00" \
--duration "03:00" --recur-every "Month Second Sunday" --time-zone "Central Standard Time"
```

2.7 Phase 6 — Managed PostgreSQL

Do not use Paperclip's embedded PostgreSQL for a PHI-bearing deployment. Managed Postgres gives you customer-managed key encryption, automated backups, and point-in-time restore that Azure demonstrates rather than you proving.

```
az postgres flexible-server create -g $RG -n $PGSRV -l $LOC \
--tier Burstable --sku-name Standard_B2s --version 16 \
--storage-size 64 --backup-retention 35 --geo-redundant-backup Enabled \
--public-access None --vnet $VNET --subnet $SUBNET_PE \
--admin-user pclipdba --admin-password "$(openssl rand -base64 32)" \
--tags $TAG

az postgres flexible-server parameter set -g $RG -s $PGSRV \
--name require_secure_transport --value ON
az postgres flexible-server parameter set -g $RG -s $PGSRV \
--name log_connections --value ON
az postgres flexible-server parameter set -g $RG -s $PGSRV \
--name log_disconnections --value ON
```

Attach the customer-managed key through the portal or --key/--identity flags once a user-assigned managed identity with Key Vault Crypto Service Encryption User exists.

Store the admin password directly into Key Vault — it must never appear in shell history or a file:

```
az keyvault secret set --vault-name $KV -n pg-admin-password --value "<generated>"
```

Verify: az postgres flexible-server show -g \$RG -n \$PGSRV --query "network.publicNetworkAccess" returns Disabled.

2.8 Phase 7 — Host hardening

Connect over Tailscale (2.10) or temporarily via Azure Bastion, then:

```
# Mount the encrypted data disk
sudo mkfs.ext4 /dev/disk/azure/scsi1/lun0
sudo mkdir -p /srv/paperclip
echo "/dev/disk/azure/scsi1/lun0 /srv/paperclip ext4 defaults,nosuid 0 2" \
| sudo tee -a /etc/fstab
sudo mount -a
```

```
sudo apt update && sudo apt full-upgrade -y
sudo apt install -y unattended-upgrades fail2ban auditd audispd-plugins chrony
sudo dpkg-reconfigure -plow unattended-upgrades
sudo systemctl enable --now fail2ban auditd chrony
sudo timedatectl set-ntp true
```

SSH policy in `/etc/ssh/sshd_config`:

```
PermitRootLogin no
PasswordAuthentication no
KbdInteractiveAuthentication no
AllowUsers pclipadmin
MaxAuthTries 3
ClientAliveInterval 300
ClientAliveCountMax 2
X11Forwarding no
```

Audit rules in `/etc/audit/rules.d/pclip.rules`:

```
-w /usr/bin/dockerd -k docker
-w /var/lib/docker -k docker
-w /etc/docker -k docker
-w /srv/paperclip -p wa -k phi-data
-w /etc/ssh/sshd_config -p wa -k sshd
-a always,exit -F arch=b64 -S execve -F euid=0 -k rootcmd
sudo augenrules --load && sudo systemctl restart auditd
```

Verify: `sudo aureport --summary` produces output; `sudo sshd -T | grep -E "permitrootlogin|passwordauth"` shows both disabled.

2.9 Phase 8 — Docker and container lockdown

```
sudo apt install -y ca-certificates curl gnupg
sudo install -m 0755 -d /etc/apt/keyrings
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo gpg --dearmor -o
/etc/apt/keyrings/docker.gpg
sudo chmod a+r /etc/apt/keyrings/docker.gpg
echo "deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.gpg]
https://download.docker.com/linux/ubuntu $(. /etc/os-release && echo $VERSION_CODENAME) stable" \
| sudo tee /etc/apt/sources.list.d/docker.list
sudo apt update && sudo apt install -y docker-ce docker-ce-cli containerd.io docker-compose-plugin
```

`/etc/docker/daemon.json`:

```
{
  "userns-remap": "default",
  "live-restore": true,
  "no-new-privileges": true,
  "icc": false,
  "userland-proxy": false,
  "log-driver": "json-file",
  "log-opts": { "max-size": "50m", "max-file": "5" }
}

sudo systemctl restart docker
REMAP=$(awk -F: '/^dockremap/{print $2}' /etc/subuid)
sudo mkdir -p /srv/paperclip/data
sudo chown -R "$REMAP":"$REMAP" /srv/paperclip/data
```

Build Paperclip from a pinned tag. Never latest.

```
cd ~ && git clone https://github.com/paperclipai/paperclip.git && cd paperclip
git checkout tags/v2026.831.1
sudo docker build --target production -t paperclip:v2026.831.1 .
sudo docker image inspect paperclip:v2026.831.1 --format '{{.Id}}'
```

--target production is mandatory. The Dockerfile declares a cloud stage after production; omitting the target silently builds the wrong variant.

Scan before running:

```
sudo docker run --rm -v /var/run/docker.sock:/var/run/docker.sock \
  aquasec/trivy:latest image --severity HIGH,CRITICAL --exit-code 1 \
  paperclip:v2026.831.1
```

Gate: no CRITICAL findings reach production. Accepted findings get an owner, rating, and due date in the risk register.

2.10 Phase 9 — Private access and the Open Dental link

```
curl -fsSL https://tailscale.com/install.sh | sh
sudo tailscale up --ssh --hostname=pclip-prod --accept-routes
sudo tailscale serve --bg 3100
```

tailscale serve publishes Paperclip at <https://pclip-prod.<tailnet>.ts.net> with a valid certificate, reachable only from the tailnet.

On the practice's Open Dental host, install Tailscale as a subnet router advertising only that machine's address — not the whole practice LAN.

In the Tailscale admin console:

- **Tailnet lock enabled**, with the Azure VM and each admin device signed. This removes the coordination server from the trust path.
- **Device approval on**.
- **ACL** permitting pclip-admins → pclip-prod:3100, and pclip-prod → Open Dental host on the API port only. Default deny everything else.
- **SSO with MFA** against the practice's identity provider.
- **Key expiry enabled** so departed staff devices drop off automatically.

Tailscale is a supporting safeguard for the transmission security requirement, not a compliance product. It stores only metadata, has no services in scope for HIPAA, and does not execute BAAs with US healthcare customers — its own DPA states it is not a business associate. That position holds precisely because it carries data it cannot decrypt. Document that reasoning in the vendor file rather than filing a BAA that does not exist.

Verify: from an admin device, `curl https://pclip-prod.<tailnet>.ts.net/api/health` returns `{"status":"ok"}`. From a non-tailnet host, the same request must fail to resolve or connect.

2.11 Phase 10 — Deploy Paperclip

/srv/paperclip/docker-compose.yml:

```
services:
  paperclip:
    image: paperclip:v2026.831.1
    container_name: paperclip
    restart: unless-stopped
    ports:
      - "127.0.0.1:3100:3100"
    environment:
      HOST: "0.0.0.0"
      PAPERCLIP_HOME: "/paperclip"
```

```

PAPERCLIP_DEPLOYMENT_MODE: "authenticated"
PAPERCLIP_DEPLOYMENT_EXPOSURE: "private"
PAPERCLIP_BIND: "custom"
PAPERCLIP_BIND_HOST: "0.0.0.0"
PAPERCLIP_AUTH_PUBLIC_BASE_URL: "https://pclip-prod.<tailnet>.ts.net"
PAPERCLIP_ALLOWED_HOSTNAMES: "pclip-prod.<tailnet>.ts.net"
PAPERCLIP_SECRETS_STRICT_MODE: "true"
DATABASE_URL_FILE: "/run/secrets/pgurl"
secrets:
  - pgurl
volumes:
  - /srv/paperclip/data:/paperclip
cap_drop: [ ALL ]
cap_add: [ CHOWN, SETUID, SETGID, DAC_OVERRIDE ]
security_opt:
  - no-new-privileges:true
  - apparmor=docker-default
read_only: true
tmpfs:
  - /tmp:rw,noexec,nosuid,nodev,size=512m
  - /run:rw,nosuid,nodev,size=64m
pids_limit: 512
mem_limit: 6g
cpus: 1.8
healthcheck:
  test: [ "CMD", "curl", "-fsS", "http://127.0.0.1:3100/api/health" ]
  interval: 30s
  timeout: 5s
  retries: 3
logging:
  driver: json-file
  options: { max-size: "50m", max-file: "5" }

secrets:
  pgurl:
    file: /srv/paperclip/.pgurl

sudo install -m 600 /dev/null /srv/paperclip/.pgurl
# write the connection string; it never enters shell history
sudo tee /srv/paperclip/.pgurl > /dev/null
sudo docker compose -f /srv/paperclip/docker-compose.yml up -d

```

Bootstrap the first account:

```
sudo docker exec -it paperclip npx paperclipai auth bootstrap-ceo
```

Open the printed invite URL from a tailnet device and create the account.

Verify: login screen loads over HTTPS at the tailnet hostname; `docker inspect paperclip --format '{{.HostConfig.ReadonlyRootfs}}'` returns true.

2.12 Phase 11 — Language model under BAA

Confirm before connecting anything:

1. Anthropic BAA executed by the organization's Primary Owner.
2. HIPAA enabled on the organization (1P API requires contacting Anthropic Sales after signing).
3. **Zero data retention enabled.** Paperclip's `claude_local` adapter runs the Claude Code CLI, which is covered under the BAA only with ZDR active.
4. **Agents pinned to non-Covered models** — Opus 5, Sonnet 5, or Haiku 4.5. Covered Models (Table 5.1, Mythos-class) require a 30-day retention minimum and are therefore unavailable with ZDR. A model change is a compliance change, not a tuning change; it goes through change control.

5. **No MCP connectors carrying PHI.** Third-party data flows through MCP are not covered by Anthropic's BAA even on a covered plan. If a connector is ever needed, it requires its own BAA and its own entry in the vendor file.

2.13 Phase 12 — Secrets

Every credential lives in Paperclip's company Secrets store, backed by the Azure-encrypted volume, with the master key mirrored into Key Vault. Nothing sensitive appears in an environment variable, a compose file, a comment, a ticket, or a shell profile.

PAPERCLIP_SECRETS_STRICT_MODE=true enforces this server-side: keys matching *_API_KEY, *_TOKEN, *_SECRET are rejected as inline plaintext when persisting agent adapter env, project env, hire-agent approvals, and company imports.

Bind each key by reference:

```
{
  "env": {
    "ANTHROPIC_API_KEY": {
      "type": "secret_ref",
      "secretId": "<uuid-from-secrets-ui>",
      "version": "latest"
    },
    "OPENDENTAL_CUSTOMER_KEY": {
      "type": "secret_ref",
      "secretId": "<uuid-from-secrets-ui>",
      "version": "latest"
    }
  }
}
```

Three traps:

- **Strict mode does not cover `config.llm.apiKey`.** That field is plain text with no secret-ref support. Leave it empty and bind per-agent. `paperclipai configure --section llm` writes plaintext there regardless of strict mode.
- **Strict mode blocks new inline values but does not migrate existing ones.** On any inherited install, run `pnpm secrets:migrate-inline-env` (dry run, then `--apply`) and grep the config yourself.
- **Back up the master key immediately.** It sits at `/srv/paperclip/data/instances/default/secrets/master.key` and encrypts every secret in the instance.

```
sudo az keyvault secret set --vault-name $KV -n paperclip-master-key \
  --file /srv/paperclip/data/instances/default/secrets/master.key --encoding base64
```

Rotate every 90 days, on staff departure, and immediately on suspected exposure. Log each rotation with date, actor, and reason.

2.14 Phase 13 — Open Dental integration and minimum necessary

The query layer is the compliance control. Agents must never hold general query access to the practice database. Build a narrow read-only service that exposes exactly one operation.

Requirements for that service:

- Runs on the VM, authenticates to Open Dental with credentials resolved from the Secrets store at runtime.

- Executes one parameterized query: patients with a treatment-planned procedure over a value threshold, not completed, last visit between 6 and 24 months ago, not deceased, not flagged do-not-contact.
- Returns **only** these fields: patient ID, name, one preferred contact method, proposed procedure description, proposed value, last visit date.
- Rejects any request for additional fields at the code level. Not by prompt instruction — by schema.
- Writes an access log entry per execution: timestamp, invoking agent, record count, requesting ticket ID.
- Enforces a per-run record cap. A query returning the entire patient base is a defect, not a large result.

```
Open Dental → [narrow read service: field allowlist + row cap + access log]
               → Paperclip ticket (minimum necessary fields only)
               → agent drafts reengagement message
               → Compliance Reviewer agent checks
               → human approves → practice sends from its own system
```

Agents do not send. They draft. A staff member reviews and sends from the practice's existing patient communication system. This keeps the outbound channel inside systems that already have consent records and delivery logs.

De-identification: what it is and is not for

Under this design agents legitimately see identified PHI, under BAA, at minimum necessary scope. De-identification is therefore not the access control — the query allowlist is. Where de-identification still applies:

- **Analytics and reporting.** Aggregate performance reporting ("recovered \$X across Y patients") must use de-identified or aggregate data. Never ship a named patient list into a dashboard.
- **Testing and development.** Non-production environments get synthetic data only. Never a copy of production.
- **Anything leaving the environment.** Vendor demos, case studies, MSP internal reporting: de-identified or nothing.

When de-identification is required, use Safe Harbor: remove all 18 identifier categories (names; geographic subdivisions below state level, ZIP beyond the first three digits under the population rule; all date elements more specific than year and any age over 89; phone, fax, email; SSN; medical record numbers; health plan and account numbers; certificate and license numbers; vehicle and device identifiers; URLs; IP addresses; biometrics; full-face images; and any other unique identifying number, characteristic, or code), and confirm no actual knowledge that the remainder could identify anyone. Run it as a scripted, deterministic step with human verification recorded — **never as an LLM redacting text in the course of its own reasoning**, which produces no verifiable artifact.

The marketing boundary

A message to a patient about a treatment their own dentist proposed is a **treatment communication**, which is excluded from HIPAA's definition of marketing and requires no separate authorization. That is what makes this use case viable.

It stops being a treatment communication, and requires prior written patient authorization under §164.508(a)(3), if:

- A third party pays the practice to include their product or service in the message.
- The content shifts from the specific proposed treatment toward promoting other products, whitening packages, financing offers, or membership plans.

- The list is repurposed for a general campaign rather than individualized follow-up.

The CEO agent does not adjudicate this line. Every message template goes to the Compliance Reviewer, and anything ambiguous goes to the human Privacy Officer. Separately, text messaging invokes TCPA consent rules — a different statute — so verify how the practice captures phone consent before any SMS channel is enabled.

2.15 Phase 14 — Company configuration and agent governance

- **Authenticated mode, private exposure.** Already set.
- **Board approval required for hires.** Leave the default. Every new agent is a new actor with new data access.
- **Per-agent budgets** at roughly 1.5× expected monthly spend. Agents stop at 100%.
- **One named account per human.** No shared logins — that is the unique user identification requirement at §164.312(a)(2)(i), and the only way the audit log means anything.

The Compliance Reviewer agent reviews every plan touching patient data, communication, or marketing before execution; drafts SOPs; escalates ambiguity rather than resolving it; and assembles the monthly evidence pack. It may block. The CEO may not override a block.

It is not the Privacy Officer. That role, and the Security Official role, are named humans required by §164.530(a)(1) and §164.308(a)(2), carrying personal accountability. Agent-drafted policies are drafts until a human approves and signs them.

Practice policy manual: load it into company context, PHI-scrubbed and version-dated, so agents cite actual practice policy rather than generic HIPAA text. Re-sync whenever the practice revises it.

CEO agent standing instructions

Paste into the CEO agent's role description:

You operate on behalf of a dental practice that is a HIPAA covered entity. Patient information in this environment is real and legally protected. Your governing constraint is not efficiency.

Minimum necessary. You may access only patient name, one contact method, proposed treatment description and value, and last visit date, and only through the approved query service. You may not request clinical notes, radiographs, insurance identifiers, Social Security numbers, guarantor financials, or payment information. If a task appears to require them, stop and escalate — do not look for another route to the data.

You draft; you do not send. No patient communication leaves this environment through you. Prepare it, route it for review, and hand it to a human to send from the practice's own system.

Marketing. Messages about a treatment the patient's own dentist proposed are treatment communications and are permitted. Anything that promotes additional products or services, or that a third party has paid to include, requires prior written patient authorization. You do not decide which category a message falls into. Every template goes to the Compliance Reviewer before use.

De-identification. You do not perform it. Stripping identifiers inside your own reasoning is not a valid procedure. Any reporting that leaves this environment must use aggregate figures with no patient-level detail.

Containment. Do not copy patient information between tickets, into artifacts, into logs, or into external tools. Each piece of patient data stays in the ticket it arrived in.

Secrets. Never write an API key, token, or password into a ticket, plan, artifact, or comment. If you encounter one in plaintext, report it as a security incident.

Hiring. You may propose agents. You may not hire them. Every proposal must state what data the agent will access and what external services it will reach.

When rules conflict, or a request sits on the line, stop and escalate. Escalating is always correct. Guessing never is.

2.16 Phase 15 — Monitoring, logging, backup

```
az monitor log-analytics workspace create -g $RG -n $LAW -l $LOC \
  --retention-time 730 --tags $TAG

az monitor diagnostic-settings create -n diag-kv \
  --resource $(az keyvault show -n $KV -g $RG --query id -o tsv) \
  --workspace $(az monitor log-analytics workspace show -g $RG -n $LAW --query id -o tsv) \
  --logs '["category":"AuditEvent","enabled":true]'
```

Enable Microsoft Defender for Cloud and Defender for Containers on the subscription. Forward the VM's auditd, Docker daemon, and Paperclip application logs into the workspace, and apply an immutability policy so retained logs cannot be altered.

Backup with soft delete and immutability:

```
az dataprotection backup-vault create -g $RG -n $BVAULT -l $LOC \
  --storage-setting "[{type:'GeoRedundant',datastore-type:'VaultStore'}]" \
  --immutability-state Locked --tags $TAG
```

Retain 35 days point-in-time for the database, 90 days for VM snapshots. **Test a restore before go-live, and quarterly thereafter.** A backup you have never restored is a hypothesis.

Alerts that page someone: failed sign-ins above threshold, Key Vault access denials, container restart loops, egress to a non-allowlisted destination, query service returning above the record cap, any agent budget hitting 100%.

2.17 Phase 16 — Validation before go-live

Run every check. Record results in the deployment ticket.

#	Check	Pass condition
1	SEV-SNP active	dmesg \ grep -i sev confirms
2	OS disk CMK encryption	DES type is ConfidentialVmEncryptedWithCustomerKey
3	No public IP anywhere	az network public-ip list -g \$RG returns empty
4	Database not public	publicNetworkAccess is Disabled
5	Key Vault firewalled	defaultAction is Deny, private endpoint healthy
6	Paperclip unreachable off-tailnet	connection fails from an external host
7	Container hardening	read-only rootfs true, caps dropped, limits set

#	Check	Pass condition
8	Strict secrets mode	inline key insertion is rejected
9	config.llm.apiKey empty	confirmed in config
10	Model pin	non-Covered model in every agent config
11	ZDR active	confirmed with Anthropic
12	Query allowlist	request for an out-of-scope field is rejected by the service
13	Record cap	oversized query is refused
14	Access logging	query service writes an entry per run
15	Restore test	database and VM restored successfully
16	Log immutability	policy applied and verified
17	Maintenance reboot	stack returns healthy after forced restart
18	MFA enforcement	non-MFA sign-in blocked
19	Tailnet lock	unsigned node cannot join
20	Safeguards Summary	delivered, signed, filed

2.18 Ongoing operations

Monthly — evidence pack: Paperclip activity log export, approvals granted, agent roster changes, model pins, secret rotations, image digest and scan results, patch status, Tailscale device changes, query service access log summary, any PHI boundary exceptions with disposition.

Quarterly — access review across Azure RBAC, Paperclip accounts, tailnet devices, and the docker group. Restore test. Re-scan image and host. Review the egress allowlist. Confirm the loaded policy manual matches the practice's signed version.

Annually — update the practice's security risk analysis to include this environment (required under §164.308(a)(1)(ii)(A), and the most common finding in enforcement actions against small practices). Re-confirm BAA coverage against any newly enabled service. Run a tabletop exercise: "an agent pasted a clinical note into a ticket," walked end to end including the breach assessment decision.

Six-year retention on risk analyses, signed policies, BAAs, training records, access reviews, incident reports, change tickets, and evidence packs, per §164.316(b)(2)(i).

References

- Azure VM sizes: <https://learn.microsoft.com/en-us/azure/virtual-machines/sizes/overview>

- DCasv6 series: <https://learn.microsoft.com/en-us/azure/virtual-machines/sizes/general-purpose/dcasv6-series>
- Confidential VM with CLI: <https://learn.microsoft.com/en-us/azure/confidential-computing/quick-create-confidential-vm-azure-cli>
- Azure HIPAA/HITECH offering: <https://learn.microsoft.com/en-us/azure/compliance/offerings/offering-hipaa-us>
- Anthropic BAA: <https://privacy.claude.com/en/articles/8114513-business-associate-agreements-baa-for-commercial-customers>
- Anthropic Covered Models: <https://support.claude.com/en/articles/15425695-covered-models>
- Paperclip deployment docs: <https://docs.paperclip.ing/reference/deploy/overview/>
- Tailscale security posture: <https://tailscale.com/security>
- Docker HIPAA checklist: <https://www.accountablehq.com/post/docker-hipaa-compliance-guide-requirements-best-practices-and-secure-setup-checklist>
- HHS business associate guidance: <https://www.hhs.gov/hipaa/for-professionals/privacy/guidance/business-associates/index.html>

Technical and operational guidance. Not legal advice. The practice's privacy counsel should review the minimum necessary determination in 2.14 and the marketing boundary before go-live.