



VIRTUAL APPLIANCE · RELEASE 1.0

ORE Hammer Pentest Setup and Usage Guide

Deploy the appliance in VMware vSphere, complete setup, and run your first authorized web application penetration test, from the OVA to a signed report.

OVA ore-pentest-1.0.0-amd64

Guest Hardened Ubuntu 24.04

Hypervisor vSphere / ESXi, Workstation, Fusion

Access HTTPS on port 8443

Contents

01 What you are installing

02 Before you start

03 Deploy the OVA

04 First boot and the console

05 The setup wizard

06 First sign-in and MFA

07 Finish setting up

08 How testing works

09 Run your first assessment

10 Review, sign off, report

11 Operating the appliance

12 Troubleshooting

13 Quick reference

CHAPTER 01

What you are installing

ORE Hammer Pentest is an agentic web application penetration testing platform that runs entirely inside your network, as a single virtual appliance.

The appliance tests the websites and APIs you authorize. It signs in as your test users, works out what each one can reach, attempts real attacks inside the limits you set, and proves each issue before reporting it. AI reasoning and fixed checks both look for issues independently. Every finding records how it was proven, and a named reviewer signs off the report.

DELIVERED AS**One OVA**

Signed release, images pre-loaded, hardened guest.

RUNS**On-premises**

Findings, evidence and credentials stay on the appliance.

AI**Your provider**

Connects to the LLM API you approve, with your key.

OUTPUT**Signed report**

PDF, HTML, Markdown and JSON, with reviewer sign-off.

What is inside

Inside the VM, the appliance runs these services. Only the HTTPS gateway is reachable from your network:

SERVICE	ROLE
Gateway	The TLS front door on port 8443. It is the only published port.
Web UI	The application you use in the browser.
API	Authentication, applications, findings, reports and settings.
Worker	Runs assessments: the testing agent and its tools.
Database and queue	SurrealDB and Redis, on the appliance's data disk.

**Integrity by design**

Every image in the appliance is checked against a signed release manifest each time it starts. Licences and offline vulnerability-intelligence updates are verified against keys pinned in the release. The appliance never trusts a key that arrives with an update.

CHAPTER 02

Before you start

Have these ready before you deploy. Most setup problems come from a missing item on this page.

Get the appliance and your licence

RaptiCore reviews each request. Once it is approved, one email brings everything you need: the licence, the download link and this guide.

1 Request the appliance.

Fill in the form at www.raptiCore.com/products/orehammer/on-prem with a business email address, your use case and how many applications you plan to test. You get an acknowledgement straight away.

2 Receive the approval email.

It has your licence attached as two files, `license.json` and `license.sig`, made for your organization with its expiry date and limits (if your mail gateway removes attachments, use the email's [Download licence](#) link). It also has your download link, valid for 7 days; each click starts a download that stays valid for 24 hours and can be resumed.

3 Download four files.

`ore-pentest-1.0.0-amd64.ova` (about 3 GB), `SHA256SUMS`, `SHA256SUMS.sig` and `release-public.pem`. The last three are also on the page above, which needs no form, so you can get them over a second channel.

4 Keep the licence files safe.

You install them in the setup wizard ([Chapter 05](#)). The appliance checks them offline and never contacts RaptiCore. For a renewal or more applications, contact RaptiCore; the new licence keeps your organization and data.

Choose a size

Pick the size when you deploy. Each execution cell is one worker that runs one assessment at a time. Memory is fully reserved for the VM.

SIZE	VCPU	MEMORY	ASSESSMENTS AT ONCE	CHOOSE IT WHEN
std	12	16 GiB	1	Most teams. Start here.
std-2	16	24 GiB	2	Two assessments regularly run at the same time.
std-3	20	32 GiB	3	Larger programmes with three parallel assessments.

Disks: a 40 GiB system disk and a 250 GiB data disk, which holds the database, captured evidence and backups. For `std-2` and `std-3`, grow the data disk in vSphere (see [Operating the appliance](#)).



Do not under-size the VM

The appliance checks its capacity every time it starts and refuses to run on less than its size needs. A `std` VM with less than 16 GiB of memory stops at first boot with a capacity message.

Network

DIRECTION	WHAT	WHY
Inbound	TCP 8443 (HTTPS) from your users	The web interface. Nothing else needs to be reachable.
Outbound	HTTPS and DNS to your approved LLM API host	AI identification and validation.
Outbound	HTTP(S) to the applications you will test	The assessments themselves.
Outbound	NTP	Licence and certificate checks need the right time.
Optional	Inbound TCP 80	Only for a Let's Encrypt certificate on a publicly resolvable name.

Checklist

- The OVA, `SHA256SUMS`, `SHA256SUMS.sig` and `release-public.pem`, verified as in [Chapter 03](#).
- Your licence: two files, `license.json` and `license.sig`, issued by Rapticore.
- A hostname for the appliance (for example `pentest.corp.example`), plus an IP address if you will not use DHCP.
- DNS and NTP servers the VM can reach.
- The first administrator's name and email address.
- An LLM provider account and API key. You enter the key in the web interface after setup, never in a file.
- Written authorization to test each application, and test accounts for the applications that have sign-in, two per role.
- Optional: a TLS certificate for the hostname from your CA. You can also install it after setup.

CHAPTER 03

Deploy the OVA

Import the appliance into vSphere, pick its size and fill in its properties. The steps are the same in the vSphere Client and in Workstation or Fusion.

1

Verify the download

Check that the release key is Rapticore's, that it signed the checksum file, and that the OVA matches it; if any check fails, do not deploy. You need OpenSSL 3 (included in macOS and Linux, and in Git for Windows). The fingerprint printed here is the reference: `d9da8dfced99776b650847226ed1f795a3b1caebb5b3f57c5e26ccbf52b64745`. The same value appears on www.rapticore.com/products/orehammer/on-prem; if the two ever differ, stop and contact Rapticore.

```
# 1. Must print the fingerprint above
openssl pkey -pubin -in release-public.pem \
  -outform DER | shasum -a 256
# 2. Must print "Signature Verified Successfully"
openssl pkeyutl -verify -pubin -inkey release-public.pem \
  -rawin -in SHA256SUMS -sigfile SHA256SUMS.sig
# 3. Must print "ore-pentest-1.0.0-amd64.ova: OK"
grep ore-pentest-1.0.0-amd64.ova SHA256SUMS | shasum -a 256 -c
```

On Windows, `Get-FileHash .\ore-pentest-1.0.0-amd64.ova` gives the value to compare with `SHA256SUMS` for step 3.

2

Deploy OVF Template

In the vSphere Client, right-click a cluster or host and choose [Deploy OVF Template](#), then select the OVA file. The OVA in this release is not code-signed, so vSphere shows it without a publisher certificate. The signed checksum from step 1 is your proof that the file came from Rapticore unchanged.

3

Choose the configuration

Select `std`, `std-2` or `std-3` (see [Choose a size](#)).

4

Choose storage and network

Put both disks on storage with room to grow, and attach the VM to the network your users and target applications can reach.

5

Fill in the properties

Complete the deployment properties described below. No secrets are entered here; the appliance generates its own.

6

Power on

Open the VM's [Web Console](#) and watch first boot ([Chapter 04](#)).

Deployment properties

PROPERTY	REQUIRED	WHAT TO ENTER
<code>ore.hostname</code>	Yes	The VM's hostname.
<code>ore.network.mode</code>	Yes	<code>dhcp</code> or <code>static</code> .
<code>ore.network.ip</code> , <code>prefix</code> , <code>gateway</code>	If static	Address, prefix length (for example <code>24</code>) and default gateway.
<code>ore.network.dns</code> , <code>search</code>	Yes	DNS servers and search domains.
<code>ore.ntp</code>	Yes	NTP servers, comma-separated. Licence and certificate checks depend on the right time.
<code>ore.appliance_url</code>	Yes	The address users will type, for example <code>https://pentest.corp.example:8443</code> .
<code>ore.https_port</code>	No	HTTPS port. Default <code>8443</code> .
<code>ore.admin.email</code> , <code>ore.admin.name</code>	Yes	The first administrator.
<code>ore.license</code>	No	Lets you install the licence at deployment instead of in the setup wizard.
<code>ore.ai.allowed_hosts</code> , <code>ore.ai.external</code>	No	Restrict or change AI egress. Hosted providers are allowed by default in this release.
<code>ore.integrations</code>	No	Enable outbound ticketing, SIEM, email and alert delivery. Default off.
<code>ore.ssh.enabled</code> , <code>ore.ssh.authorized_key</code>	No	Support access over SSH, by key only. Default off. See Support access .



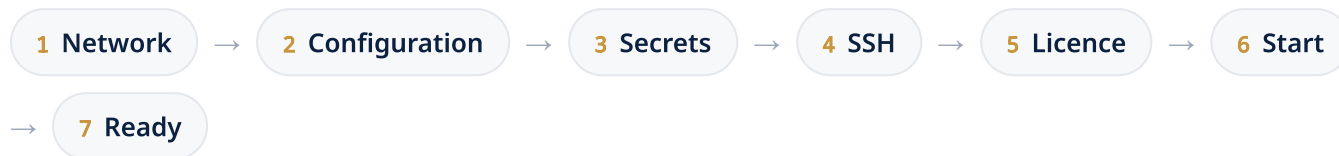
Recommended: enable support access

Commands such as backup, setup-code and MFA reset run on the appliance itself. Setting `ore.ssh.enabled` with one OpenSSH public key creates the `ore-support` account, which can sign in with that key only. You can't add it later without redeploying.

CHAPTER 04

First boot and the console

On first power-on, the appliance configures itself from your properties, generates its own secrets and starts the services. It takes a few minutes. Watch it on the VM console.



Each step prints a line on the console. When first boot finishes, the console switches to the appliance status screen. Everything you need for setup is on it:

```

ORE Hammer Pentest  ore-pentest-1.0.0

Web:      https://pentest.corp.example:8443
Certificate: SHA-256 fingerprint of the web certificate
Licensing: fingerprint of the pinned licensing key

Setup:    waiting for the first administrator
Setup code: XXXX-XXXX-XXXX-XXXX
Open https://pentest.corp.example:8443/setup and enter it.
It is valid for 24 hours and stops working once setup is done.

Licence:  not installed - enter it in the setup wizard

Enter: refresh    N: new setup code
  
```

- **The setup code** is one-time and valid for 24 hours. Press **N** on the console for a new one; the old one stops working.
- **The certificate fingerprint** lets you confirm, in the browser, that you are talking to this appliance.
- Once setup is complete, the code is no longer shown and the setup page closes for good.



If first boot stops

A failed step prints `FAILED at <step>` with the reason, then *Fix the cause, then reboot*. The usual cause is a VM with less memory than its size needs. Fix it and reboot; first boot resumes where it stopped. See [Troubleshooting](#).

CHAPTER 05

The setup wizard

Three short steps in the browser: prove you are at the console, install the licence, and create the first administrator.

1

Open the setup page

Browse to the URL shown on the console, for example

`https://pentest.corp.example:8443/setup`. The browser warns about the certificate because the appliance starts with a self-signed one. Check that the certificate's SHA-256 fingerprint matches the console, then continue.

2

Enter the setup code

Type the code from the console. A wrong or expired code is refused; get a fresh one with `N` on the console.

3

Upload the licence

Choose `license.json` and `license.sig`. The appliance verifies them against the licensing key pinned in the release.

4

Create the first administrator

Enter a full name, email address and password. The password needs at least 12 characters, must not be a commonly used password, and must not contain your name or email.

i

"Licence is not active yet"

A licence starts at its issue time. If the appliance's clock is behind, it treats a newly issued licence as not yet valid. Check that the NTP servers in `ore.ntp` are reachable, then try again.

CHAPTER 06

First sign-in and MFA

The appliance requires multi-factor authentication from the start, so the first sign-in goes straight to enrolment.

- 1 Sign in**
at the appliance URL with the administrator you just created.
- 2 Enrol an authenticator app**
(any TOTP app) by scanning the QR code and entering a code from it.
- 3 Store the recovery codes.**
They are shown once. Keep them somewhere safe, apart from the authenticator.

Nothing else opens until enrolment is complete. An administrator can relax the MFA requirement later under [Settings → Security](#) , but we recommend keeping it on.

CHAPTER 07

Finish setting up

The **Finish setting up** checklist on the Overview (and under [Settings → Appliance](#)) lists what is left. An item the appliance could not measure reads *Not checked*, never done.

ITEM	WHAT TO DO	WHERE
Licence	Done in the wizard. Renew here when the banner says it is due.	Settings → Appliance
Multi-factor authentication	Done at first sign-in.	Settings → My Profile
AI provider (<i>required</i>)	Add your LLM provider, model and API key, then run Test key and model .	Settings → AI Configuration
Web certificate	Replace the self-signed certificate.	Settings → Web certificate
First application	Register an application you are authorized to test.	Web App Pentest → Applications
Backup	Take the first backup.	On the appliance (see Backup)

Connect your AI provider

AI validation is a core part of how the appliance decides what is real. Without a provider, assessments still run, but their findings are held as *AI inconclusive* rather than presented as AI-validated.

- 1 Open**
[Settings → AI Configuration](#) .
- 2 Choose the provider.**
 OpenAI, Anthropic, Grok/xAI, Kimi/Moonshot, GLM/Z.AI, DeepSeek, or a customer-hosted OpenAI-compatible endpoint. Then enter the exact model ID.
- 3 Paste the API key.**
 It is write-only: it is encrypted on the appliance and never shown again.
- 4 Test and save.**
 Run [Test key and model](#) , then [Save and activate](#) . When the saved key passes, the checklist marks the AI provider done.

Add a secondary provider (recommended). It takes over only when the primary keeps failing. Each provider needs the key that provider issued; a key already saved for one provider is refused for another.

1

Save the secondary's own key.

Under **Provider** , choose the secondary provider, paste the key it issued, test it and save.

2

Switch back and select it.

Choose your primary provider again, pick the secondary under **Secondary provider** and save.

3

Test the secondary.

Run **Test secondary provider** to check its stored key and model.

**Keys belong in the UI only**

Never put provider keys in files, tickets or chat. Test both providers after saving: if the primary's key is rejected, every assessment runs on the secondary alone.

Install a trusted web certificate

Under **Settings → Web certificate** , choose one of:

- **Upload** a certificate chain and private key from your CA.
- **Generate a key and signing request** on the appliance, have your CA sign it, then upload the certificate.
- **Let's Encrypt**, if the hostname resolves publicly and inbound port 80 reaches the appliance.

Add your team

Under **Settings → Members** , choose **Add user** and enter a full name, email and role. The appliance shows a one-time temporary password. Hand it over through a channel you trust; no mail server is needed. The user picks their own password at first sign-in, and the temporary password expires after 7 days. For a locked-out member, use **Issue temporary password** on their row. Every account counts against your licensed user limit.

CHAPTER 08

How testing works

Four ideas explain everything you see in the product: applications, logins, depth, and how a finding was proven.

APPLICATION

What you may test

A website or API, who authorized testing, and the limits the agent must stay inside.

LOGIN

Who to test as

A test account the agent signs in with, so it can test access between users and roles.

DEPTH

How far to go

From a read-only look to a full authorized penetration test.

PROOF

How sure we are

Shown on every finding, separately from its severity.

Two independent methods

AI reasoning and fixed checks each look for issues on their own, and neither is treated as the authority over the other. Fixed checks give discovery, controlled observation and reproducible replay; AI finds what fixed rules miss and validates what was observed. A finding from fixed checks is never labelled AI-validated unless AI actually validated it.

Severity and proof are separate

Severity says how bad an issue is. **How it was proven** says how sure we are. They never share a label, so a critical finding that is only a signature match cannot be mistaken for one that was reproduced.



Not proven is not absent

A finding the automation could not conclusively validate stays in the report at its full rating until a reviewer decides it. Automation can raise a rating but never lowers one.

PROOF STATE	MEANING
Validated	Independently validated, proof-backed.
Inherited	A duplicate published on a validated finding's decision, with its own proof.
AI-validated	A real issue the AI validated from captured behaviour; no deterministic test has reproduced it yet.
Deterministic-only	A signature match where AI validation was not run. Deliberately not equivalent to a validated result.
Control gap	A configuration or hardening control observed absent: reported for compliance, not as exploitable risk.
Needs review	An effect was observed; a reviewer decides.
AI inconclusive	AI could not confirm or refute it. Recorded as inconclusive, never as a clean pass.
False positive	Excluded, with the reason.

CHAPTER 09

Run your first assessment

Register an application, add logins, choose how to test, and watch it run. The in-product guide under [Web App Pentest → Getting started](#) follows the same steps and shows where you are.

Step 1 · Add an application

Go to [Web App Pentest → Applications](#) and choose **Add application**.

FIELD	WHAT TO ENTER
Name and host	The host to test, for example <code>app.example.com</code> . A domain also covers its subdomains.
Authorization	Who authorized the test and on what basis. It is kept with the record as a statement of fact.
Scope limits	Allowed ports, allowed path prefixes and excluded paths. Anything outside them is not tested.
Environment and criticality	Used to weigh risk. A flaw in production counts for more than one in a test copy.
Schedule (optional)	Recurring assessments. Scheduled runs are read-only inside the application's limits.

The agent checks these limits before it sends any active request, and a target outside them cannot even be queued.

Step 2 · Add logins

Most serious web flaws are about access: one user reading another's data, or an ordinary user doing something only an administrator should. The agent can test that only if it can sign in as more than one user.

LOGINS	WHAT THE AGENT CAN TEST	CHOOSE THIS WHEN
None	Only what a visitor can see.	A first look, or a public site.
One	What one signed-in user can reach.	You only have one test account so far.
Two of the same role	Whether one user can reach another's data. This is the minimum for access-control testing.	Any application with user accounts.
One more per extra role	Whether a lower role can reach what a higher role can.	The application has roles such as administrator or manager.

- **How it signs in:** *Browser capture* opens a live browser window that you sign in to yourself. Form login, bearer token, API key, basic auth and cookie also work.
- **Profile ID and role:** give each login a short ID and a role such as `user` or `admin`, and a tenant if the application has tenants.

- **What each role may do:** describe what the role should and should not reach. It tells the agent what a breach looks like.
- **Save logins for later:** choose **Save these logins for future assessments**, or use the application's **Access & scope** tab, so retests and schedules sign in without retyping.



Use test accounts, not real people's

Saved credentials are encrypted with the appliance's key, are never shown again, and are left out of exports. Each extra login adds requests and run time, so add the roles that matter and stop there.

Step 3 · Choose how to test

DEPTH

DEPTH	WHAT IT DOES	CHOOSE IT FOR	IT NEEDS
Safe baseline	Looks at the application without signing in, and changes nothing.	A first look, or an application you cannot safely write to.	Nothing extra.
Authenticated application test	Signs in with your logins and tests what each user can reach, including whether one user can reach another's data. It may make safe writes.	Any application with user accounts.	At least two logins, and a short-lived run authorization.
Full authorized penetration test	Everything in the authenticated test, plus active vulnerability scanning and live exploit attempts. Each exploit is re-run independently before it counts. It can change data.	An application you own, ideally a test or staging copy.	Logins, and a short-lived run authorization.

Depths that can write need a run authorization. For a registered application, the Access step's **Authorize this run** button issues one for an hour.

AI VALIDATION

MODE	WHAT IT DOES
AI-assisted (recommended)	AI and fixed checks independently discover candidates; persisted proof, AI review and impact validation decide what is published.
AI-led (larger AI budgets)	Expands AI-led discovery and verification budgets. The same evidence, review, impact and publication controls still apply.
Fixed checks only (no AI validation)	Proven results are reported as deterministic-only, limited-assurance outcomes, clearly labelled in the app and the report.



A good first run

The **Authenticated application test**, **AI-assisted**, with **two logins**, against a test or staging copy of the application.

Step 4 · Run it and watch

Start from [Applications](#) or [Start assessment](#). The form has three steps: Scope, Access and Run. Several assessments can run at once, one per execution cell, each on its own application.

1 Recon

2 Authenticate

3 Hypothesize

4 Targeted tests

5 Sweep

6 Chain

7 Methodology report

8 Probing

9 Exploit validation

10 Review

WHERE	WHAT YOU SEE
Overview	What is running now.
Assessments	Every run, live ones first. Open one to see its stage.
Activity tab	What the agent plans, does, observes and verifies, as it happens.
Needs your input	A panel above the tabs when the run is waiting on you. Answer it and the run continues.
Stopping a run	Cancel it from the run, or from its row in Applications.

A time box or capacity limit never fails an assessment. If time runs out, the run still finishes with review, a report and a coverage ledger that lists what was skipped.

CHAPTER 10

Review, sign off, report

A person decides each critical and high finding, a named reviewer signs, and the report carries that sign-off.

Decide each finding

Open a finding from the **Findings** tab to see its evidence: the request and response, how it was replayed, and why it was flagged. Then record a decision:

DECISION	EFFECT
Confirm	Keeps it as a real finding. It then reads <i>reviewer-confirmed</i> , never AI-validated.
Source review required	Keeps it in the report at full severity until a source review settles it.
False positive	Removes it from the report body, with a reason, into the appendix.
Exclude	A deliberate exclusion, such as an accepted risk, with a reason, listed in the appendix.

[Web App Pentest → Reviews](#) lists every application whose latest assessment still has critical or high findings waiting on a reviewer. The **Coverage** tab shows what the agent discovered and how far it got, so you know what a clean result does and does not cover.

Sign off

On the assessment's **Report** tab, the reviewer states that they reviewed the critical and high findings and accept the outcomes as they stand. Sign-off does not claim the application is secure, and it does not make any finding proven.

- **Who:** an organization administrator, or a user given review permission.
- **When:** the assessment is complete, validation has finished, and nothing critical or high is waiting on a reviewer. The Report tab says what is holding it up.
- **After changes:** change any finding after signing and the sign-off reads *Out of date* until it is signed again.

Get the report

FORMAT	USE IT FOR
PDF or HTML	Sharing with someone who was not in the room. Issued by RaptiCore, with the reviewer named under "Reviewed and signed by".
Markdown	Pasting into a ticket.
JSON	Other tools.
Audit-ready report	The full document, with scope, methodology, findings and signature lines.

Findings export A CSV or PDF of findings, from the Findings tab.

Final exports are available once every unresolved finding has a decision; before that you can render a watermarked draft. Every export says whether it is *Signed*, *Out of date* or *Not signed*. After fixes, use **Retest** on the assessment to run it again with the same settings.

CHAPTER 11

Operating the appliance

Routine care: licence renewal, backups, offline intelligence updates, account recovery and support access.

Support access

Appliance commands run as the `ore-support` account over SSH. The account exists only if `ore.ssh.enabled` and `ore.ssh.authorized_key` were set at deployment. It signs in with that key only, and every `sudo` is logged in the journal and audit log.

```
ssh ore-support@pentest.corp.example
sudo ore-pentest status      # services and health
sudo ore-pentest preflight   # run every start-up check
```

Licence renewal

Administrators see a banner from 30 days before expiry. Install the renewal under [Settings → Appliance](#) by uploading the new `license.json` and `license.sig`. A renewal must name the same organization. After expiry, sign-in, existing findings and the renewal page keep working, but assessments stop until a renewed licence is installed.

Backup and restore

```
sudo ore-pentest backup /var/lib/ore-pentest/backups/2026-11-01 # the directory r
sudo ore-pentest restore /path/to/backup RESTORE                # destructive: ch
```

- A backup holds the database export, captured evidence, configuration and licence material, with integrity hashes. Treat it as sensitive, and copy it off the appliance.
- The appliance's secrets are not included. Keep them under your secret handling, because the restored data is encrypted with them.
- Restore replaces the database and queue, never deletes captured evidence, and needs the confirmation word `RESTORE`.
- [Settings → Appliance](#) shows the time of the last backup made with this command.

Offline vulnerability-intelligence updates

RaptiCore publishes signed CVE intelligence packs. Transfer a pack to the appliance and import it:

```
sudo ore-pentest import-cve-pack /path/to/extracted-pack
```

The pack is verified against the key pinned in the release, never against a key carried inside the update. It installs atomically, keeps a rollback copy, and restarts the worker. An older pack than the one installed is refused.

Account recovery

- **Lost authenticator:** sign in with a recovery code, or ask an administrator to reset MFA under [Settings → Security](#).
- **No administrator can sign in:** `sudo ore-pentest reset-mfa admin@example.com` clears that user's MFA and signs them out everywhere. They enrol again at their next sign-in.
- **New setup code:** press `N` on the console, or run `sudo ore-pentest setup-code`. This works only until setup is complete.

Growing the data disk

For `std-2` and `std-3`, enlarge the data disk (disk 2) in vSphere, then extend the volume from a support session:

```
sudo pvresize /dev/sdb && sudo lvextend -r -l +100%FREE ore-data/data
```

CHAPTER 12

Troubleshooting

The appliance refuses to run in an unsafe or under-sized state, and always says why. Start from the message.

YOU SEE	CAUSE	FIX
Console: FAILED at start ... requires at least 12 CPU threads and 16 GiB	The VM has less CPU or memory than its size needs.	Give the VM its full size (memory fully reserved) and reboot. First boot resumes.
Console: container memory limits total ... but this host leaves ...	Memory available to the guest is below what the services are configured for.	Give the VM its full memory. Only lower the service memory limits if RaptiCore support advises it.
Setup: licence is not active yet	The appliance clock is behind the licence's issue time.	Make sure the NTP servers in <code>ore.ntp</code> are reachable, then retry.
Download link no longer works	The emailed link is valid for 7 days; each click gives a download valid for 24 hours.	Click the link in the approval email again. After 7 days, ask RaptiCore for a new link; your licence stays valid.
Signature or checksum check fails	An incomplete or altered download, or files from different releases.	Download all four files again. Do not deploy until all three checks pass; contact RaptiCore if they still fail.
Setup: the setup code is incorrect	The code was mistyped, expired, or replaced by a newer one.	Press <code>N</code> on the console for a new code.
Browser certificate warning	The appliance still has its self-signed certificate.	Check the fingerprint against the console, then install a trusted certificate (Chapter 07).
Findings marked AI inconclusive	No AI provider configured, or the provider was unavailable.	Configure or fix the provider under Settings → AI Configuration, then run Test key and model (and Test secondary provider).
Banner: assessments are stopped	The licence has expired.	Install the renewal under Settings → Appliance.
A target cannot be queued	It is outside the application's registered host, ports or paths, or under an excluded path.	Adjust the application's scope limits if the test is authorized.



Contacting RaptiCore support

Include the release ID from the console (`ore-pentest-1.0.0`), the exact message, and, if support access is enabled, the output of `sudo ore-pentest preflight` . Never send licence private material, provider keys or appliance secrets.

CHAPTER 13

Quick reference

The essentials on one page.

WEB INTERFACE

:8443

https://<appliance>:8443

FIRST SETUP

/setup

Code on the VM console, valid 24 h.

SUPPORT ACCOUNT

ore-support

SSH by key, if enabled at deployment.

DEFAULT SIZE

12 vCPU · 16 GiB

std; std-2 and std-3 for parallel runs.

Where things are

TASK	WHERE
Register an application	Web App Pentest → Applications → Add application
Start an assessment	Start assessment (Overview, Applications or Assessments)
Watch a run	Web App Pentest → Assessments → open the run → Activity
Decide findings	The assessment's Findings tab; Web App Pentest → Reviews
Sign off and export	The assessment's Report tab
All findings across applications	Actions → Findings
Add users	Settings → Members → Add user
AI provider	Settings → AI Configuration
Licence and setup checklist	Settings → Appliance
Methodology and OWASP reference	Settings → Help & Docs, and the in-product reference

Appliance commands

COMMAND	DOES
<code>sudo ore-pentest status</code>	Shows services and health.
<code>sudo ore-pentest preflight</code>	Runs every start-up check and prints PASS or the reason for a failure.
<code>sudo ore-pentest setup-code</code>	Issues a new one-time setup code (until setup is complete).
<code>sudo ore-pentest backup <new-dir></code>	Takes a full backup.
<code>sudo ore-pentest restore <dir> RESTORE</code>	Restores a backup (destructive).
<code>sudo ore-pentest import-cve-pack <dir></code>	Imports a signed intelligence update.
<code>sudo ore-pentest reset-mfa <email></code>	Resets a user's MFA from the appliance.