Revert, Cleanup & Command Reference

Revert, Cleanup & Command Reference

Revert the service tag

Repeat the PWRD_EN procedure and set the tag back to `XXXXXXX`. Express becomes valid again; Enterprise fails `LIC008`. The Enterprise XML can be re-imported after setting the tag to `XXXXXXX` again.

Clean up the stale Express license (optional)

```
racadm license view
racadm license delete -t 2
```

Transaction 2 is the old Express license. Deleting it removes the device-level `LIC008` warning. Keep it if you may revert the tag.

Key commands

```
racadm getsysinfo
racadm getsvctag
racadm license view
racadm license delete -t <id>
racadm remoteimage -s
racadm closessn -a
```

SSH (iDRAC CLI is `clpd`, prompt `/admin1->`):

```
ssh -c aes128-ctr -o MACs=hmac-sha2-256 -o Compression=no operator@192.0.2.10
```

Web/API login and import are wrapped in `idrac8/xxe/` helpers; the DTD server must listen on **port 80** if the XXE route is retried.

Repository map

```
idrac8/
  RESULT.md                     final result + revert
  VULNS.md                      avenue analysis
  licenses/
    iDRAC8_Enterprise_REDACTED.xml   working license
    iDRAC7_Enterprise_9QYZF42.xml  donor (LIC010)
    iDRAC8_Enterprise_Extended_Eval_License.xml  eval (LIC009)
  xxe/
    README.md
    build_duration.py / check_duration.py
    serve.py / smtp_probe.py / idrac_ssh.py
    forged_duration*.xml, donor_utf16*.xml
    dtd_dur_verifier.dtd, dtd_dur_parser.dtd
```

Timeline

1. Forged license → `LIC017`.
2. iDRAC7 donor → `LIC010`.
3. UTF-16 DOCTYPE bypass → accepted (`LIC010` with the donor).
4. Internal parameter entity → works; external HTTP/file entity → `LIC017`.
5. Found port-80 behaviour of the iDRAC HTTP client.
6. 240-day eval → `LIC009`.

7. Found leaked iDRAC8 license (`XXXXXXX`) → `LIC008`.
8. Service-tag change to `XXXXXXX` → import → **Enterprise perpetual** `OK`.

Revision #3
Created 2026-11-20 02:41:06 UTC by Zac
Updated 2026-11-20 03:00:47 UTC by Zac