Role-Based Device Authentication Protocol (RBDAP)

Author: Mr Jan Jeltes,
JELTES Ecosystems Group,
Australia, 8 July 2025

1. Overview

The Role-Based Device Authentication Protocol (RBDAP) enables cryptographically secure, decentralised device authentication and role assertion on Layer 2/3 networks without relying on DHCP or centralised dynamic policy engines. Devices assert their identity and role through signed challenge-response handshakes validated by switches or routers against a local key registry.

2. Objectives

- Prevent spoofing, replay, and unauthorised device access
- Allow autonomous device role declaration
- Enforce policy based on cryptographically verified role assignments
- Operate in isolated or air-gapped environments
- Remain lightweight and easy to implement on constrained devices

3. Definitions

- **Device**: Any endpoint asserting a role (e.g., sensor, server, enterprise PC)
- **Authority**: A router, switch, or network service verifying assertions
- **Registry**: A local or distributed database of authorised devices and keys

4. Components

- Static IP & MAC assignments (no DHCP)
- Role assertion over UDP broadcast or directed unicast
- Challenge-response using nonces
- Local registry containing device metadata and public keys

5. Protocol Steps

5.1. Discovery

Device broadcasts (or unicast) a role discovery request:

```
{
  "msg_type": "discover",
  "device_id": "SEN-42-ALPHA"
}
```

5.2. Challenge

Authority issues a nonce:

```
{
  "msg_type": "challenge",
  "device_id": "SEN-42-ALPHA",
  "nonce": "eD6jkl89zZ",
  "expires": 1720037288
}
```

5.3. Assertion

Device replies with signed assertion:

```
{
  "msg_type": "assert",
  "device_id": "SEN-42-ALPHA",
  "ip": "192.168.20.42",
  "mac": "00:1A:2B:3C:4D:5E",
  "role": "sensor",
  "nonce": "eD6jkl89zZ",
  "sig": "0xABCDEF..."
}
```

The signature is generated over the concatenated fields:

```
device_id + ip + mac + role + nonce
```

5.4. Verification

Authority performs:

- Signature verification using registry-stored public key
- IP/MAC/role match validation
- Nonce expiration check

If valid, the role is granted and access policies are applied.

6. Registry Format

Example:

```
{
  "SEN-42-ALPHA": {
    "role": "sensor",
    "expected_ip": "192.168.20.42",
    "expected_mac": "00:1A:2B:3C:4D:5E",
    "pubkey": "-----BEGIN PUBLIC KEY-----\nMIIBIjAN..."
  }
}
```

7. Security Considerations

- **Replay Protection**: Nonces must be single-use and time-limited
- **Trust Model**: Authority must trust its local registry (airgapped or signed)
- **Key Compromise**: Rotate keys regularly and revoke on detection

8. Policy Enforcement

Upon successful assertion, network devices apply policies based on roles, such as:

- VLAN assignment
- Firewall rules
- Routing permissions
- Rate limits or isolation rules

9. Extensibility

Future additions may include:

- Firmware version checks
- Geofencing
- Role-based metrics or reporting
- Peer device verification

10. Status

Draft 0.1 — Prepared for initial internal use only.