Objectives

[a]
The authenticity of communications sessions is protected.

SC.L2-3.13.15

System & Communications Protection

Communications Authenticity

"Protect the authenticity of communications sessions."

Key Discussion Points

What It Protects Against:
Man-in-the-middle attacks, session hijacking, and insertion of false information — all require that one or both parties cannot verify the other's identity.

Mutual Authentication:
The guide explicitly recommends mutual authentication (mTLS) as the most secure approach — both parties verify each other, not just the server verifying the client.

TLS with Valid Certs:
TLS with a valid certificate from a trusted CA is the primary implementation — both the protocol version and the certificate must be current and correctly configured.

Session vs. Packet:
This control operates at the session level — it establishes trust for the entire session duration, not just individual packets or messages.

Assessment Methods

EXAMINE

System and communications protection policy; procedures addressing session authenticity; system security plan; system design documentation; system configuration settings; system audit logs.

INTERVIEW

System or network administrators; personnel with information security responsibilities.

TEST

Mechanisms supporting or implementing session authenticity.

SC.L2-3.13.15

SYSTEM & COMMS PROTECTION — Communications Authenticity

Plain English

What this control is really saying:

A man-in-the-middle attack can intercept a session between two parties, impersonate both ends, and read or modify everything exchanged — without either side knowing. This control requires that communications sessions be protected against that attack, against session hijacking, and against insertion of false information. The practical implementation is mutual authentication using valid certificates and current TLS configuration.

How it is used:

- Web servers and VPN gateways use valid TLS certificates from trusted CAs — certificates are renewed before expiration and the renewal process is tracked in the SSP.
- TLS configuration enforces TLS 1.2 or higher with strong cipher suites — weak protocols (SSL, TLS 1.0, TLS 1.1) and cipher suites are disabled.
- Mutual TLS (mTLS) is configured for server-to-server communications — both endpoints present and validate certificates before a session is established.
- Certificate configurations are reviewed annually and after any change — TLS configuration is verified against current best-practice baselines.

Real World Example

The Scenario

Acme Defense's CUI web portal uses TLS, but the server configuration allows TLS 1.0 and includes weak cipher suites (RC4, 3DES). The certificate expired 14 days ago and has not been renewed.

What the assessor finds

An assessor tests the TLS configuration using a scanning tool — the expired certificate fails authenticity validation and the server accepts a TLS 1.0 downgrade. A proof-of-concept confirms a MITM interception is feasible on the same network segment.

SPRS Score Impact

3.13.15 carries a point value of 1. An expired certificate provides no authenticity protection — and a weak TLS configuration can be actively exploited to intercept or modify session content. Both must be addressed.

What Good Looks Like

Valid TLS certificates from trusted CAs on all CUI-facing endpoints, TLS 1.2 or higher enforced with strong cipher suites, weak protocols disabled, certificate expiration tracked and renewed proactively, mTLS for server-to-server sessions where appropriate.

Common Gaps

What assessors actually find in the field:

✗ **Expired TLS certificate**

The TLS certificate on a CUI web application has expired — browsers display trust warnings and the certificate is no longer providing authenticity protection.

✗ **Weak TLS configuration**

The server supports TLS 1.0 and weak cipher suites — a downgrade attack can force use of vulnerable protocols that allow session interception.

✗ **Self-signed certificates**

Internal systems use self-signed certificates — there is no chain of trust and clients cannot verify server identity.

✗ **No certificate tracking**

No inventory of certificates exists — expired certificates are discovered when browsers display warnings rather than from proactive management.

✗ **One-way TLS only**

Only server authentication is implemented — the server does not verify the client identity, allowing session impersonation by unauthorized clients.