Internet Draft   Jesse Walker
Expiration: March 2002   Amol Kulkarni
File: draft-ietf-rap-cops-tls-01.txt   Intel Corp.

                          COPS Over TLS

                 Last Updated: September 7, 2001

Status of this Memo

    This document is an Internet-Draft and is in full conformance with
    all provisions of Section 10 of RFC2026.

    Internet-Drafts are working documents of the Internet Engineering
    Task Force (IETF), its areas, and its working groups. Note that
    other groups may also distribute working documents as Internet-
    Drafts.

    Internet-Drafts are draft documents valid for a maximum of six
    months and may be updated, replaced, or obsoleted by other documents
    at any time. It is inappropriate to use Internet-Drafts as
    reference material or to cite them other than as "work in progress."

    The list of current Internet-Drafts can be accessed at
    http://www.ietf.org/ietf/1id-abstracts.txt

    The list of Internet-Draft Shadow Directories can be accessed at
    http://www.ietf.org/shadow.html.

Conventions used in this document

    The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
    "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in
    this document are to be interpreted as described in [RFC-2119].

Abstract

    This memo describes how to use TLS to secure COPS connections over
    the Internet.

    Please send comments on this document to the rap@ops.ietf.org
    mailing list.

Table Of Contents

1. **Introduction**

    COPS [COPS] was designed to distribute clear-text policy information
    from a centralized Policy Decision Point (PDP) to a set of Policy
    Enforcement Points (PEP) in the Internet. COPS provides its own
    security mechanisms to protect the per-hop integrity of the deployed
    policy. However, the use of COPS for sensitive applications such as
    some types of security policy distribution requires additional
    security measures, such as data privacy. This is because some
    organizations find it necessary to hide some or all of their security
    policies, e.g., because policy distribution to devices such as mobile
    platforms can cross domain boundaries.

    TLS [TLS] was designed to provide channel-oriented security. TLS
    standardizes SSL and may be used with any connection-oriented
    service. TLS provides mechanisms for both one- and two-way
    authentication, dynamic session keying, and data stream privacy and
    integrity.

    This document describes how to use COPS over TLS. "COPS over TLS" is
    abbreviated COPS/TLS.

2. **COPS Over TLS**

    COPS/TLS is very simple: use COPS over TLS exactly as you would use
    COPS over TCP.

2.1. **Connection Initiation**

    The system acting as the PEP also acts as the TLS client. This system
    initiates a connection to the PDP to the secure COPS port. When this
    succeeds, the PEP system sends the TLS ClientHello to begin the TLS
    handshake. When the TLS handshake completes, the PEP MAY initiate the
    first COPS message. All COPS data MUST be sent as TLS "application
    data". Normal COPS behavior follows.

    All PEP implementations of COPS/TLS MUST support an access control
    mechanism to identify authorized PDPs. This requirement provides a
    level of assurance that the policy arriving at the PEP is actually
    valid. The access control mechanism implemented is outside the scope
    of this document. PEP implementations SHOULD require the use of this
    access control mechanism for operation of COPS over TLS. When access
    control is enabled, the PEP implementation MUST NOT initiate COPS/TLS
    connections to systems not authorized as PDPs by the access control
    mechanism.

    Similarly, PDP COPS/TLS implementations MUST support an access
    control mechanism permitting them to restrict their services to

authorized PEP systems only. Implementations MUST NOT require the use
of an access control mechanism at the PDP, however, as organizations
might not consider as sensitive the types of policy being deployed,
and therefore do not need to incur the expense of managing

credentials for the PEP systems. However, if access controls are
used, the PDP implementation MUST terminate COPS/TLS connections from
unauthorized PEP systems and log an error if an auditable logging
mechanism is present.

2.2. Connection Closure

TLS provides facilities to securely close its connections. Reception
of a valid closure alert assures an implementation that no further
data will arrive on that connection. The TLS specification requires
TLS implementations to initiate a closure alert exchange before
closing a connection. It permits TLS implementations to close
connections without waiting to receive closure alerts from the peer,
provided they send their own first. TLS allows implementations to
reuse the session in this case, but COPS/TLS makes no use of this
capability.

Note that a premature close does not call into question the security
of the data already received, but simply indicates that subsequent
data might have been truncated. Because TLS is oblivious to COPS
message boundaries, it is necessary to examine the COPS data itself
(specifically the Message header) to determine whether truncation
occurred.

2.2.1. PEP System Behavior

PEP implementations MUST treat premature closes as errors and any
data received as potentially truncated. The COPS protocol allows the
PEP system to find out whether truncation took place. A PEP system
detecting an incomplete close SHOULD recover gracefully.

PEP systems MUST send a closure alert before closing the connection.
Clients unprepared to receive any more data MAY choose not to wait
for the PDP system's closure alert and simply close the connection,
thus generating an incomplete close on the PDP side.

2.2.2. PDP System Behavior

COPS permits a PEP to close the connection at any time, and requires
PDPs to recover gracefully. In particular, PDPs SHOULD be prepared to
receive an incomplete close from the PEP, since a PEP often shuts
down for operational reasons unrelated to the transfer of policy
information between the PEP and PDP.

   Implementation note: The PDP ordinarily expects to be able to
   signal end of data by closing the connection. However, the PEP
   may have already sent the closure alert and dropped the
   connection.

PDP systems MUST attempt to initiate an exchange of closure alerts with the PEP system before closing the connection. PDP systems MAY close the connection after sending the closure alert, thus generating an incomplete close on the PEP side.

2.3. Port Number

The first data a PDP expects to receive from the PEP is a Client-Open message. The first data a TLS server (and hence a COPS/TLS server) expects to receive is the ClientHello. Consequently, COPS/TLS runs over a separate port in order to distinguish it from COPS alone. When COPS/TLS runs over a TCP/IP connection, the default TCP port at the PDP is TBD. The PEP may use any TCP port. This does not preclude COPS/TLS from running over another transport. TLS only presumes a reliable connection-oriented data stream.

3. Endpoint Identification and Access Control

Implementations of COPS/TLS MUST use X.509 v3 certificates conforming to [PKIX] to identify PDP and PEP systems. COPS/TLS systems MUST perform certificate verification processing conforming to [PKIX].

If a subjectAltName extension of type dNSName or iPAddress is present in the PDP's certificate, that MUST be used as the PDP identity. Otherwise, the most specific Common Name field in the Subject field of the certificate MUST be used.

Matching is performed using the matching rules specified by [PKIX]. If more than one identity of a given type is present in the certificate (e.g. more than one dNSName name, a match in any one of the set is considered acceptable.), the COPS system uses the first name to match, except as noted below in the IP address checking requirements. Names may contain the wildcard character * which is considered to match any single domain name component or component fragment. For example, *.a.com matches foo.a.com but not bar.foo.a.com. f*.com matches foo.com but not bar.com.

3.1. PDP Identity

Generally, COPS/TLS requests are generated by the PEP consulting bootstrap policy information identifying authorized PDPs. As a consequence, the hostname or IP address for the PDP is known to the PEP. How this bootstrap policy information arrives at the PEP is outside the scope of this document. However, all PEP implementations MUST provide a mechanism to securely deliver or configure the bootstrap policy. In particular, all PEP implementations MUST support a mechanism to securely acquire the signing certificate of the authorized certificate authorities issuing PDP certificates, and MUST support a mechanism to securely acquire an access control list or filter identifying its set of authorized PDPs.

PEP implementations that participate in multiple domains, such as those on mobile platforms, MAY use different certificate authorities and access control lists in each domain.

Organizations may choose to deliver some or all of the bootstrap
policy configuration from an untrusted source, such as DHCP. In this

circumstance, COPS over TLS provides no protection from attack when
this untrusted source is compromised.

If the PDP hostname or IP address is available via the access control
mechanism, the PEP MUST check it against the PDP's identity as
presented in the PDP's TLS Certificate message.

In some cases the bootstrap policy will identify the authorized PDP
only by an IP address of the PDP system. In this case, the
subjectAltName MUST be present in the certificate, and it MUST
include an iPAdress format matching the expected name of the policy
server.

If the hostname of the PDP does not match the identity in the
certificate, a PEP on a user oriented system MUST either notify the
user (PEP systems MAY afford the user the opportunity to continue
with the connection in any case) or terminate the connection with a
bad certificate error. PEPs on unattended systems MUST log the error
to an appropriate audit log (if available) and MUST terminate the
connection (with a bad certificate error). Unattended PEP systems MAY
provide a configuration setting that disables this check, but then
MUST provide a setting which enables it.

3.2. PEP Identity

When PEP systems are not access controlled, the PDP need have no
external knowledge of what the PEP's identity ought to be and so
checks are neither possible nor necessary. In this case, there is no
requirement for PEP systems to register with a certificate authority,
and COPS over TLS uses one-way authentication, of the PDP to the PEP.

When PEP systems are access controlled, PEPs must be PKI clients in
the sense of [PKIX]. In this case, COPS over TLS uses two-way
authentication, and the PDP MUST perform the same identity checks for
the PEPs as described above for the PDP.

When access controls are in effect at the PDP, PDP implementations
MUST have a mechanism to securely acquire the signing certificates of
the certificate authorities issuing certificates to any of the PEPs
they support.

4. IANA Considerations

COPS over TLS uses a separate TCP port from COPS. IANA should assign
the value TBD to this port.

5. Security Considerations

This entire document concerns security.

[6](#). Acknowledgements

This document freely plagiarizes and adapts Eric Rescorla's similar document draft-ietf-tls-http-xx.txt that specifies how HTTP runs over TLS. Discussions with David Durham and Ylian Sainte-Hillaire also lead to improvements in this document.

7. References

[COPS] Durham, D., Boyle, J., Cohen, R., Herzog, R., Rajan, R., Sastry, A., "The COPS (Common Open Policy Service) Protocol", RFC 2748, January 200.

[PKIX] Housley, R., Ford, W., Polk, W., Solo, D., "Internet Public Key Infrastructure: Part I: X.509 Certificate and CRL Profile", RFC 2459, January 1999.

[RFC2026] Bradner, S., "The Internet Standards Process - Revision 3", RFC 2026, October 1996

[RFC2119] Bradner, S., "Key Words for use in RFCs to indicate Requirement Levels", RFC 2119, March 1997.

[TLS] Dierks, T., Allen, C., "The TLS Protocol", RFC2246, January 1999.

8. Author Addresses

Jesse R. Walker
Intel Corporation
2111 N.E. 25th Avenue
Hillsboro, OR 97214
USA
jesse.walker@intel.com

Amol Kulkarni
Intel Corporation
JF3-206
2111 N.E. 25th Avenue
Hillsboro, OR 97214
USA
amol.kulkarni@intel.com