Endpoint Security

Introduction

The aim of this paper is to demonstrate how endpoint security status can be used to enable access within a Collaboration Oriented Architecture. There is no intent to duplicate the work by the trusted computing group[1] and it is expected that implementation of solutions suggested by this paper will rely on much of this work.

Problem

In a de-perimeterised environment secure collaboration between two devices is achieved by their security association rather than trying to keep the network pure, thus there is a need to be able to validate that the endpoint is secure, and assert that status as part of any transaction.

With most current end-point checking solutions there is an over-reliance on network level authentication; that is, at the point at which you are able to assert a level of control over a device connecting to the network. This approach is flawed in a de-perimeterised environment, where the end-point device connects to a "foreign" network (say a public IP, or Wi-Fi hotspot) and then connected directly to devices inside your organisation using inherently secure protocols.

Thus the key question for anyone designing an end-point security solution in Collaboration Oriented Architecture is; "what triggers the checking and when do you need to do it". Any checking must extend beyond the pure Operation Systems checks to (potentially) include being able to assert the security status of applications that will be used in the transaction as well as (potentially) checking that unwanted application are not running. Such checking may be one-time, or may need to be continuous dependant on the type of collaboration and the process involved in the transaction.

Key issues for a Collaboration Oriented Architecture

Connection issues

There are two question to be asked when a system or user tries to initiate a collaborative session (distinct from "tries to connect", which implies that the check is only performed at connection initiation):

- Are they allowed to collaborate? Does the device and/or the user have the necessary credentials to make a connection (probably a bi-directional check[2]); and;
- Are they in a fit state to connect? This is a risk decision based on attributes of the system being used for the collaboration, its ability to assert its status, and the security requirements defined for the collaboration to take place.

When designing systems for use in Collaboration Oriented Architectures then the goal should be to design out as many dependencies on the collaborating systems as possible.

[1] http://www.trustedcomputinggroup.org
[2] Jericho Forum Commandment #7 - https://www.opengroup.org/jericho/commandments_v1.2.pdf

The points at which trust needs to be validated will vary on the risk and the type of transaction. Some transactions will necessitate that trust can only be established at the start of the transaction, but a more preferable solution will be to perform continuous checking, (a security heartbeat) and better still will be for the application itself to revalidate (or request revalidation of) the trust levels, potentially as the types of transaction being requested of the application changes (for example requesting a bank balance vs. transferring money).

Security status could be established via a bi-directional trust or the use of a third-party "in-the-cloud" trust-brokers service.

The addition of Trusted Platform Module (TPM)[3] aware software, or software download, has the potential to provide levels of trust/assurance; however the problem with any trust system is the potential for that software or communication to be interfered with or impersonated.

The need for secure communication

All components of the Trusted Computing Module(s) within a device, the anti-malware, the personal firewall, hardware, OS etc. need to be able to securely communicate their status.

The communication of status in a de-perimeterised environment must be vendor-neutral as no presumption can be made that a particular vendor solution will be present on either end-point.

If a third-party or trust broker is being used to assert the status of a device then the end-point(s) will need to securely report their status, logs, etc. to its parent/home trust broker.

When designing a system to assert trust, then it is important to remember that the collaboration may be multi-way, or even using broadcast protocols.

The need for remediation

Where an end-point does not meet the required level of trust then one option will be to allow (or force) that device to remediate itself, with the end-point locked-out from the transaction / collaboration process until it is remediated.

Ideally, a device that requires remediating will be locked-out of all transactions (via, say a change of rules to its personal firewall) until it has been remediated. Such lock out and remediation must function irrespective of whether the device is Intranet or Internet connected.

Rather than bar the transaction if a device does not meet the required standard, it may be possible to modify the transaction method, or access allowed, based on the evaluated risk. A negotiation process may need to take place to establish the optimal method of a mutually trusted collaboration possible between devices. The negotiation may be direct or via a trust-broker.

The calculation of risk

The calculation of risk and thus the decision to collaborate generally needs to be an automatic process that happens in the background without the user being aware that (if it is successful) it has taken place. Should the transaction be refused then the workflow in establishing that connection may involve the user in an "are you sure" choice (assuming this is a user to system transaction and not a system-to-system transaction).

Such risk decision will be based on a variety of factors pertinent to the desired transaction, including;

- The physical/geographic location of the device(s)/users(s)
- The patch status of the device

[3] Trusted Platform Module (TPM) Specifications - https://www.trustedcomputinggroup.org/specs/TPM/

- The operating system, status and patch level (programs running, registry settings, etc.)
- Whether the device can safely hold data (is the container encrypted, can such a container be killed or it's keys repudiated, etc.)
- Can it securely communicate (does the device speak the appropriate secure protocols)
- The availability of the appropriate application(s), and/or ability to downgrade the application used – see above
- The date and/or time of the transaction
- The identity of the user
- The nature of the transaction

Although some of these factors in the risk calculation can be derived, most will need to be communicated by the device in a standard, secure and open format.

Risk calculations must be dynamic and relevant to the transaction in-progress rather than a single transaction on entry to the system (or worst still, the network). Dependant on the risk, some systems and/or people may only be allowed a basic level of low-risk transaction (for example view-only access). A combination of the above factors combined with a risk calculation should define whether access is barred, allowed but limited, or full.

Challenges to the industry

Current end-point security solutions are proprietary, and generally designed to operate in homogenous, perimeterised environments. There is a need to embrace open standards such as:

- TNC specification - IF-TNCCS-SOH[4] is now available for anyone to download or implement.
- IETF NEA Network Endpoint Assessment[5] - RFC 5209[6]

Key to working in Collaboration Oriented Architectures will be the ability of the components that determine and/or demonstrate the security status of an end-point device to communicate that status in a secure way, irrespective of where that end-point device physically located or connected.

To achieve this, the key vendors in the space need to agree an open specification/standard they will all use by which such status can be securely communicated, and then prove their interoperability.

The way forward

The ability to communicate security status will allow this information to be used by endpoint and intermediate devices in automated risk calculations.

The ability to query the status of a device, directly or via third-party brokers or other certification devices or services, has the potential to allow applications, and other devices in the path (such as identity-aware firewalls) to be able to request this information to make access decisions (both network and application), and also to set granular levels of access to both functionality within applications as well as to information.

---

[4] An open standard available at https://www.trustedcomputinggroup.org/groups/network
[5] Network Endpoint Assessment working group at http://www.ietf.org/html.charters/nea-charter.html
[6] http://tools.ietf.org/html/rfc5209