COA\(^1\) Processes

Device Lifecycle Management

Introduction

In a collaborative, de-perimeterised environment, if the device needs be trusted then the management of devices becomes critical when collaborating with another device, system or service.\(^2\)

The entire lifecycle of a device needs to be managed, from the point of device creation or on-boarding\(^3\) process to register external devices as permissible to connect to other devices, systems or services, through managing the device and ensuring the software / OS environment on the device stays current, to eventually repudiating the device and potentially removing any sensitive data or configuration information.

Whether the device is a client or server, the same issues exist and while the lifecycle will be similar it will probably be managed under a more formalised process for servers.

Problem

Device life cycle management is at an embryonic stage, with corporates who run a locked down end-user computing configuration with heavily customised software management system getting close to the required level of management - but often only inside the corporate boundary, and at the cost of end user productivity - while full management when roaming the Internet is only available in a closed eco-system with correctly configured / locked-down “closed” devices.

The current status is increasingly at odds with the proliferation of types of devices, business demands for flexibility, and the conflict with operating a locked down model, and consumerisation of devices.

Key issues for a Collaborative Oriented Architecture

Need for Trust

As already mentioned, appreciating the Jericho Forum’s design principles on the need for trust\(^2\) is critical here:


\(^2\) Jericho Forum design principles #6 & #7 on the need for trust - http://www.opengroup.org/jericho/commandments_v1.2.pdf

\(^3\) Definition: “On-boarding” – The process of enrolling a device or person such that they have the capability to be recognised by the systems they need to interact with.
#6 - All people, processes, technology must have declared and transparent levels of trust for any transaction to take place

- Trust in this context is establishing understanding between contracting parties to conduct a transaction and the obligations this assigns on each party involved
- Trust models must encompass people/organisations and devices/infrastructure
- Trust level may vary by location, transaction type, user role and transactional risk

#7 - Mutual trust assurance levels must be determinable

- Devices and users must be capable of appropriate levels of (mutual) authentication for accessing systems and data
- Authentication and authorisation frameworks must support the trust model

Need for consistent Intranet / Internet management

The management of devices must function identically irrespective of whether a device is connected to the Intranet or Internet, with the operation of such management secure. All protocols involved in the management of the device must be inherently secure⁴.

Provisioning, fixes, suitable software & patches

The provisioning of new software, the de-provisioning and/or removal of old or unsuitable software, together with patch management, and configuration of software must cover all software (OS, BIOS, and application software) on the device. The provisioning mechanism must be secure and all protocols involved in the provisioning of the software must be inherently secure.

On-boarding – registration

Where a level of trust is required for a device to collaborate with other devices, systems or services, then there must be an on-boarding or registration process. This registration process may be manual (as in the case of building a new corporate PC, or a server) but ideally should be ad-hoc and not make assumptions about where the device is trying to register from.

The capability for ad-hoc on-boarding is especially important in the case of alien (unknown / non-on-boarded) devices that are required to collaborate, where the on-boarding process involves network based access. Here, mutual trust assurance levels must be determinable for the two systems (registration service, and new device) involved in the transaction⁵.

Identity of device

The device needs to be capable of positively and uniquely identifying itself to other systems in a form that cannot be subverted (thus using a MAC address is not a suitable level of authentication).

This may involve using an existing suitable key on the device, loading a key, or using software, as part of the registration process, or could be achieved by federating the device’s identity from the organisation you are collaborating with.

Suitability of device to interact

Information about the end device state and capability, combined with the (automated) risk assessment, should all affect an assessment of how suitable a device is to interact.

⁵ Jericho Forum Commandment #7, see Jericho Forum design principles, commandment #7
Factors that may dictate how a device can interact include:

- amount of memory
- presence of a particular installed application
- how the connection is being made
- the network speed and/or cost of connection.

Depending on the interaction, it may be possible to vary the method of interaction to better suit the circumstances in which the two collaborating devices find themselves, potentially with a negotiating process to find a mutually agreeable method of interaction.

**Off-boarding devices**

Devices that are on-boarded must have a complementary process to off-board them. Preferably this should be part of a continual process to ensure and re-validate the need for device-level access, especially where devices are not under your direct control.

**Recovery of data, & key repudiation**

Where devices have been off-boarded or where the access changes, then any local data on that device may need to be recovered or destroyed. Keys may need to be repudiated, and potentially any software that was loaded onto that device as part of the on-boarding and/or management process should be removed and de-licensed.

**The need to federate devices and/or applications and/or services**

Just as devices under your control are managed in an authorisation database or directory, so devices of trusted parties should be able to connect via federated access to their authorisation database, in the same way that user access can be federated.

**Endpoint Security configuration management**

Where the transaction taking place requires a level of trust in the integrity of the operating system and software then there must be a secure method of establishing the acceptability of the configuration state of that device, and a decision made about whether to allow that transaction / collaboration (see the “COA – Endpoint Security” paper) or use an alternative method of collaborating.

**Device remediation**

The ability to lock-out a device from the transaction / collaboration process until it is remediated is essential. Ideally, a device that requires remediation 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 and (dependant on the transaction) could potentially happen during the transaction.

**Challenges to the industry**

**Gateway connections**

Endpoint security checking performed at a network or perimeter entry point is generally flawed in a de-perimeterised environment.

---

In a de-perimeterised environment the communication will be (secure) protocol based rather and the action of a device or user authenticating to a network device. Thus the network connection process (and thus the ability to intercept and check the device) will have already have taken place, in the worst case on a public Wi-Fi or other network that is outside of your control.

**Vendor neutrality**

When an endpoint device is already part of a management eco-system (for example a laptop from a consulting company visiting your site one day a week) then local systems should be able to leverage this to allow access, and should even allow federated device access in the event that an established business relationship exists. End-devices, users and companies will not tolerate an increasing set of unique endpoint solutions being loaded and probably interfering as part of a local on-boarding process.

Vendors should collaborate to develop such secure standards.

**Connecting via a tunnelled connection**

The current method of using a generic (open to all protocols & services) IPSec (or SSL) tunnel to allow alien devices into your closed network is severely flawed, and highly undesirable. Such solutions have the potential open up both endpoint and network to problems that may exist at either end, and do not meet the need for inherently secure communications⁷ to reduce the attack surface and thus help provide both least privilege and defense in depth.

**The way forward**

Current device management solutions need to be enhanced to operate in a collaborative / de-perimeterised environment. Such solutions not only need to be able to support a multitude of devices, but also need the additional capability to securely communicate their status to other partners in any collaboration.

Vendors should also be working with the producers of consumer devices to identify methods by which these devices can be on-boarded to allow their operation in a collaborative environment.

Relevant applications will need to be enhanced to support the validation of device when making a connection request to their systems.

---