Luigi Catuogno, Hans Löhr, Mark Manulis, Ahmad-Reza Sadeghi, Christian Stüble, Marcel Winandy

Trusted Virtual Domains: Color Your Network

Trusted Virtual Domains (TVDs) provide a secure IT infrastructure offering a homogeneous and transparent enforcement of access control policies on data and network resources. In this article, we give an overview of the fundamental ideas and basic concepts behind TVDs, present a realization of TVDs, and discuss application scenarios.

Dr. Luigi Catuogno is responsible for network and system administration at the University of Salerno, Italy. He was a visiting researcher at Ruhr-University Bochum, Germany.
E-Mail: luicat@dia.unisa.it

Hans Löhr is research assistant at the Horst Görtz Institute for IT-Security (HGI) at Ruhr-University Bochum, Germany.
E-Mail: hans.loehr@trust.rub.de

Prof. Dr.-Ing. Mark Manulis is professor at the Center for Advanced Security Research Darmstadt (CASED) at Technical University Darmstadt, Germany.
E-Mail: manulis@informatik.tu-darmstadt.de

Prof. Dr.-Ing. Ahmad-Reza Sadeghi is professor at the Horst Görtz Institute for IT-Security (HGI) at Ruhr-University Germany.
E-Mail: ahmad.sadeghi@trust.rub.de

Christian Stüble is CTO of Sirrix AG security technologies.
E-Mail: stueble@sirrix.com

Marcel Winandy is research assistant at the Horst Görtz Institute for IT-Security (HGI) at Ruhr-University Bochum, Germany.
E-Mail: marcel.winandy@trust.rub.de

Introduction

Enterprises and governmental organizations often struggle with the problem that employees have to use IT systems for different tasks with different security requirements. They may have to deal with top-confidential data while they are also working on data and documents that are supposed to be shared with others. Employees perform different tasks under different roles, for example accessing the Internet, using intranet services, editing unclassified documents, as well as editing classified documents, such as patents. Each of these kinds of tasks has different security requirements. In security-critical environments such as government and military, classified documents are isolated by using physically separated computing platforms. However, in typical enterprise environments users perform these tasks using one computing platform providing a questionable isolation between them. Instead we can observe the opposite trend, i.e., more and more infrastructure is shared for several tasks, and sometimes even for several organizational units or even complete enterprises. For example, cloud computing offers infrastructure and services for different customers on the same hardware platforms.

While sharing IT infrastructure is cost-efficient and provides more flexibility, it increases the security problems organizations have to deal with in order to isolate data of different workflows and to fulfill confidentiality demands while data (and system) sharing is required. In addition, employees can work with mobile computing platforms which are not always under control of the organization's domain. It is also not unusual to send documents to private computers in order to work from home, and later bring the data back into the organizational domain. If private computers are not protected sufficiently, data may leak outside the organization, or malicious code enter the organization due to this data transfer.

In this context, security concerns become even more urgent when mobile storage devices are used, such as portable hard drives and USB memory sticks, which offer additional flexibility for the transport of data across multiple working locations and devices (e.g., work stations, printers, cell phones, cameras, etc.) [10]. Such an extension needs to take into account diverse security risks with regard to the data stored on the devices. For example, they can be easily lost or stolen, and consequently the confidentiality of data becomes an issue. Once left unattended by the user, mobile devices can be manipulated with the goal to breach the integrity of the data or to disseminate corrupted data or malicious code once the device is reconnected to the enterprise platform. Many security solutions for mobile

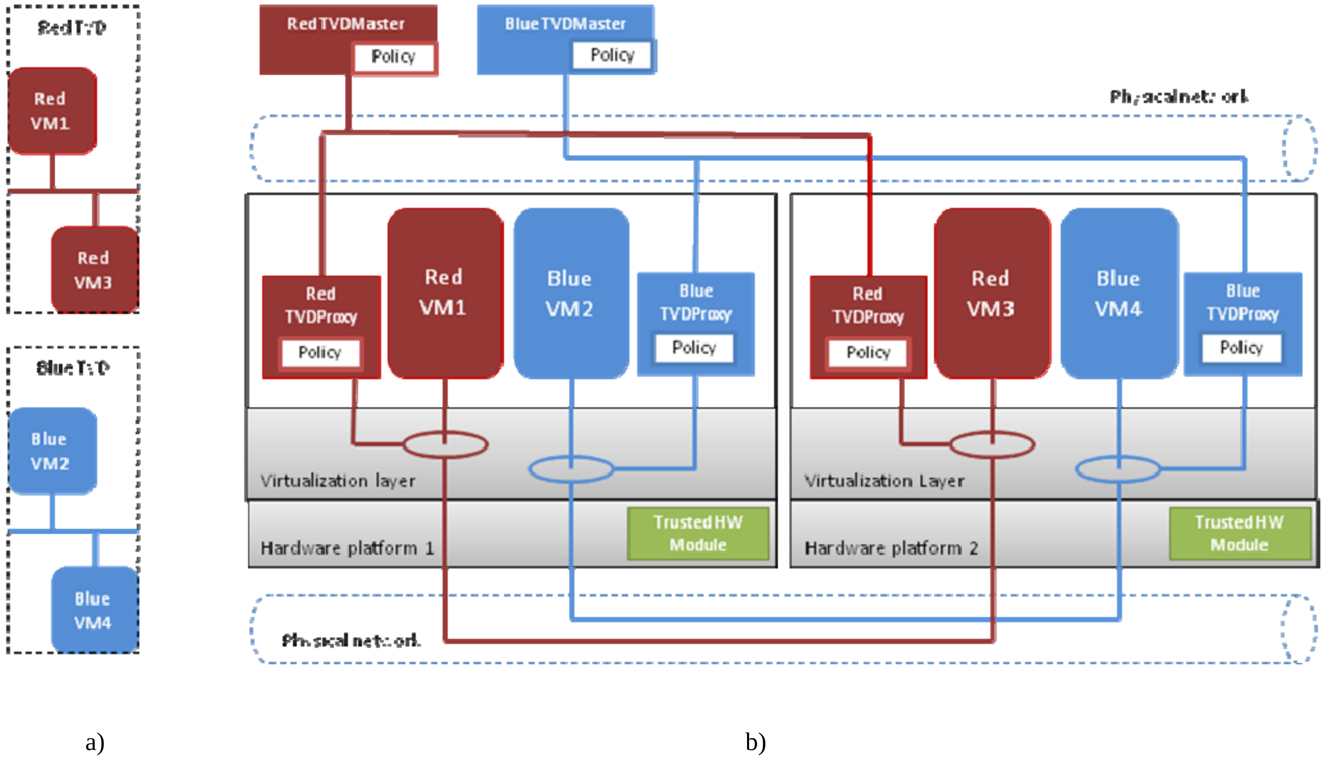

Figure 1: Overview of Trusted Virtual Domains (TVDs). *Part a) shows the logical view of two TVDs distributed over two physical machines. Part b) shows the physical deployment of the TVD components, including the TVD Master.*

storage devices adopted in practice rely on a mixture of different techniques. In fact, the choice of appropriate mechanisms is guided by trade-off between their costs and offered benefits [14, 13]. Recent surveys indicate that existing security policies vary across organizations from none to very restrictive ones disallowing those devices completely [11, 12].

But it is possible to deal with all these security concerns in a still manageable way. Trusted Virtual Domains (TVDs) are a suitable framework for the implementation of secure multi-domain / single-infrastructure computer networks like centralized data centers, where computational resources from different owners share the same physical infrastructure, or single organizational LANs that span over different offices, branches or functional areas.

Amongst the strengths of TVDs is the transparent data protection and enforcement of access control policies — platforms and users logically assigned to the same TVD can access distributed data storage, network services, and remote servers without executing any additional security protocols, while the resources belonging to different TVDs are strictly separated and, thus, remain inaccessible for the unauthorized. Moreover, data that is stored on mobile storage devices is automatically protected by encryption and can only be decrypted within the same TVD the device has been assigned to. Hence, users cannot forget to employ encryption, and data on memory sticks cannot be used outside the TVD.

In this paper, we give an overview and introduction to the concepts of TVDs, as well as examples for concrete real-world applications.

Overview of TVDs

Trusted Virtual Domains (TVDs) [6, 2] are a novel security framework for distributed multi-domain environments which leverages virtualization and trusted computing technologies. In this section we give a brief overview of the TVD concept and its features, as well as its main components and protocols.

In a virtualized environment, different applications and services together with their underlying operating systems are executed by different Virtual Machines (VMs) that share the same physical infrastructure. Each virtual machine runs in a logically isolated execution environment (which we call compartment), controlled by the underlying Virtual Machine Monitor (VMM). In such an environment, the user's work space is executed in a virtual machine.

A TVD is a set of virtual machines that trust each other, share a common security policy and enforce it independently of the particular physical platform they are running on. Moreover, the TVD infrastructure contains the VMM and the physical components (such as CPU, memory, and hardware security modules) on which the virtual machines rely to enforce the policy. In particular, the main features of TVDs and the TVD infrastructure are:

- *Isolation of execution environments.* The underlying VMM provides containment boundaries to compartments from different TVDs, allowing the execution of several different TVDs on the same physical platform.
- *Trust relationships.* A TVD policy defines which platforms (including VMM) and which virtual machines are allowed to join the TVD. For example, platforms and their virtualization layers as well as individual virtual machines can be identified via integrity measurements taken during their start-up.
- *Transparent policy enforcement.* The Virtual Machine Monitor enforces the security policy transparently from the user or any applications running within virtual machines..
- *Secure communication channels.* Virtual machines belonging to the same TVD are connected through a virtual network that can span over different platforms and that is strictly isolated by the virtual networks of other TVDs. Depending on the application scenario, different mechanisms (such as virtual private networks pro-

viding encryption) can be used to secure the communication (see below).

Figure 1 shows an example of two TVDs (a "red" and a "blue" one) that are distributed over different physical machines, and illustrates main components of the TVD architecture and their relations. The TVD policy is a set of rules that state security requirements a compartment should satisfy in order to be allowed to join the TVD (e.g., integrity measurements of the platform and VMs) and defines both intra-TVD and inter-TVD information flow policy. A special node, namely the TVD Master, logically acting as a central server, controls the access to the TVD following the admission control rules stated in the TVD policy. The TVD Proxy is a compartment that locally enforces the TVD policy on the platform it is running on. Several TVD Proxies, belonging to different TVDs can be instantiated on the same platform.

The process of TVD establishment in two steps, "deploy" and "join", is detailed in [8]: With the TVD deploy protocol, the TVD Master verifies a platform and its ability to enforce the TVD policy. Then, in the TVD join procedure, the TVD Proxy (verified by the TVD Master during the deploy phase) can verify virtual machines that are executed on the platform, and admit them to the TVD. Trusted computing technology is used to establish trust through attestation mechanisms.

For example – following the TCG approach – hash values of the software boot stack (including BIOS, bootloader, and virtualization layer as well as loaded virtual machines) are stored in and signed by a Trusted Platform Module (TPM) [9] and reported to the TVD Master during an attestation protocol. The TVD Master can reliably verify whether the reported values match the required ones of the TVD policy. Based on this, the TVD Master can implicitly rely on the enforcement mechanisms of the local platforms.[1]

Techniques to isolate and manage the virtual networks of different TVDs are given in [3]. Basically, virtual switches on each platform implement VLAN tagging for local connections[2], and secure virtual private network (VPN) for remote connections.

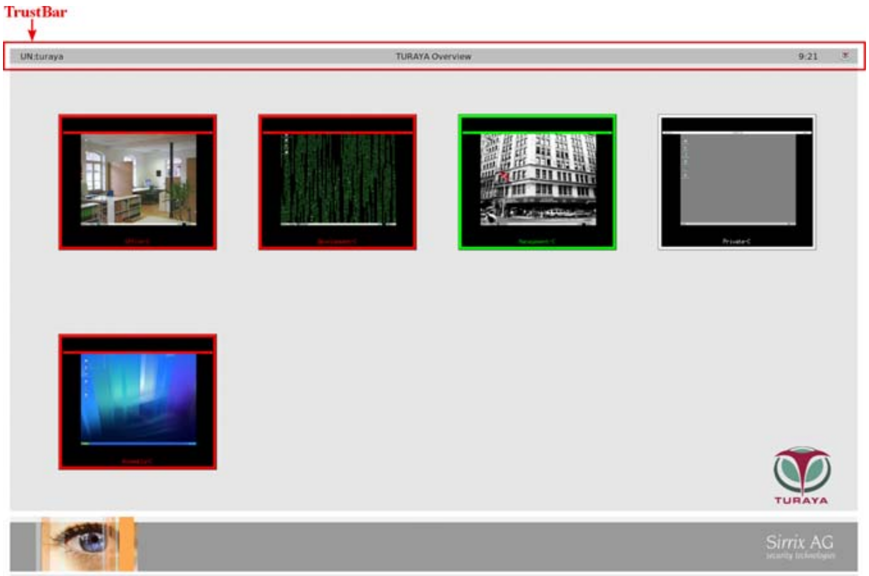

Figure 2: *Overview screen of the TrustedGUI showing five compartments. The TrustBar on the top of the screen securely shows the name of the compartment that controls the screen.*

Various applications of TVDs were already shown and discussed in the literature. One example addresses the idea of applying the TVD concept for secure information sharing [7]. Other examples are virtual data centers [1], or enterprise rights management [5] and the secure incorporation of mobile storage devices [10].

TVD Management

The leading approach of management of TVDs within both centralized Virtual Data Centers and distributed organizational networks leverages on the deployment of advanced network management technologies (e.g., the Web-based Enterprise Management [4]) that provide highly integrated tools to accomplish administration tasks.

In a TVD-enabled infrastructure, management activities span over three levels. The infrastructure level concerns maintenance of physical resources, setup and configuration of the overall logical infrastructure, and assignment of resources to the different TVDs. At domain level, administrators take care of the TVD deployment, virtual machine setup and management of policies, devices and keys. Finally, at compartment level, running applications and current users can be notified of some events, coming from the underlying platform (e.g., revocation of a VM).

At each level, administrators have an integrated management console that allows them to control all the operations under their responsibility. The administration of Virtual Data Centers with TVDs is discussed in [1].

The normal operation of a TVD requires mechanisms for membership revocation and policy updates as part of the general life cycle management. For instance, changes in resource assignment and access privileges require the modification of the currently active TVD Policy, as well as the revocation of any TVD components instantiated based on the old policy. In these cases, the TVD Master must revoke the old TVD policy and distribute the new one to all hosts where the respective TVD is deployed. Care must be taken that all hosts are notified and hosts which are off-line or otherwise ignore the update are isolated from the updated TVD. Up to now, tools to support policy management are still work in progress.

Realization of TVDs and Application Scenarios

Based on research results of different R & D projects, e.g., EMSCB[3] and OpenTC[4], the Turaya product family[5] has been developed to provide a distrib-

[1] The definition of the required integrity measurement values in the TVD policy postulates knowledge about the behavior and security properties of the corresponding software programs. In practice, this can be achieved, e.g., through independent trusted third parties who evaluate and certify products according to evaluation standards like Common Criteria.

[2] Secure communication between VMs on one platform must be provided by the VMM. Security measures of the VMs themselves (e.g., encryption) are not sufficient, because the VMM can access the internal state of the VMs, and hence circumvent such protection mechanisms.

[3] See www.emscb.de
[4] See www.opentc.net
[5] See www.sirrix.com

uted trusted environment based on TVDs.

The central component of such a trusted IT-infrastructure is the Turaya.TrustedObjectsManager (TOM), the management console of security policies and the IT infrastructure including different types of appliances such as the TrustedVPN appliance, the TrustedDesktop appliance and the TrustedMobileDesktop appliance.

TrustedObjectsManager: The central management component of a trusted infrastructure provides the user interface to define TVDs and corresponding intra-TVD and inter-TVD policies. Moreover, the TOM manages the physical infrastructure including networks, services, and appliances.

Since appliances remotely enforce a subset of the overall security policy, a permanent trusted channel [15,16] between TOM and its appliances is used for client authentication, to check their software configuration using attestation, and to upload policy changes and software updates. Finally, the TOM acts as TVD-Master by creating an independent TVD-specific root-CA for each defined TVD.

TrustedVPN: The Turaya.TrustedVPN appliance acts as trusted VPN gateway to enforce TVD policies based on connected networks. In addition, the TrustedVPN appliance acts as a gateway for software VPN clients. The only information stored permanently on a TrustedVPN is the identifying signature key protected by a Hardware Security Module (HSM), the network address, and the public certificate of the corresponding TOM. On startup, the TrustedVPN connects to the TOM which checks the appliances identity and the validity of its software configuration. If the appliance passes these tests, the TOM derives from its internal database the TVDs the appliance is allowed to connect to, invokes the appliance to create a new signature key pair for each TVD, and certifies these keys using its TVD-specific root CA key. From now on, the TrustedVPN can establish VPN tunnels to other appliances. However, if the appliance is turned off, it loses all TVD-specific credentials.

TrustedDesktop: The Turaya.TrustedDesktop appliance is a trusted communication end-point of a managed infrastructure. Based on virtualization and isolation realized by the underlying Turaya security kernel, it allows users to work in parallel with isolated compartments connected to different TVDs. Thus, TrustedDesktop enforces the intra-TVD and inter-TVD policies defined by the TOM. An important security-service of this appliance is the TrustedGUI providing a trusted path to the user (see Figure 2). The TrustBar allows users to securely identify the active compartment and its related TVD, as well as to switch between compartments and to do some local configuration management such as network configuration and the management of compartments.

Instead of preventing data flows (e.g., by disabling the USB port) and thus constrain users in doing their work, TrustedDesktop acts as an encryption layer that is transparent to the user. Two examples are the use of shared file systems such as a USB storage and Cut & Paste:

- As with a conventional GUI, user scan copy information in one compartment (e.g., select some text from a document) that they want to paste into another compartment. The TrustedGUI encrypts data pasted into a compartment if the information flow is not allowed according to the current policy, and decrypts it if the information flow is allowed. This way, two compartments that belong to the same TVD but without network access can exchange information using any communication channel (e.g., email) of an untrusted TVD.
- Information stored onto an untrusted file system, such as a USB stick, is transparently encrypted using a key bound to the TVD of the storing compartment. Thus, compartments of different TVDs can share the USB stick without violation of the security policy. Moreover, users can exchange information even between different platforms as long as an information flow between source TVD and destination TVD is allowed by the security policy.

Identical to the TrustedVPN appliance, the TrustedDesktop appliance establishes a management tunnel to the TOM using a trusted channel. The management tunnel is also used to download new compartment images from the TOM and to perform the initial user authentication.

TrustedMobileDesktop: The Turaya.TrustedMobileDesktop appliance is the mobile counterpart of TrustedDesktop and currently available as a demonstration prototype. On top of an OMAP 35xx development board including an ARM Cortex A8, two Linux compartments (running MAEMO or Android) are executed in parallel on top of a microkernel. Both Linux compartments are strongly isolated from each other. While one compartment, the "User Linux", is completely open and configurable by the user, the second compartment is part of a TVD and thus isolated using Turaya security services such as file encryption and trusted VPN client.

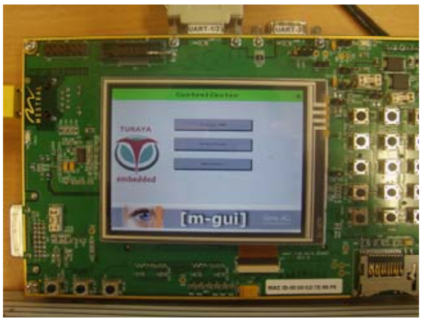

Figure 3: *The Turaya.TrustedMobileDesktop with the m-gui on an embedded platform.*

The TrustedMobileDesktop appliance also provides a simplified TrustedGUI service, the "m-gui", including a TrustBar to protect the integrity, confidentiality, and authenticity of user input and output (see Figure 3).

Enterprise Rights Management

The TVD infrastructure can be deployed in corporate environments to support various use cases. Here, we illustrate this with the example of enterprise rights management (ERM) [5].

Enterprise document workflows entail the following three requirements: Firstly, documents accessed on particular computing platforms need strict isolation and protection from other unauthorized processes or users of the platforms. Secondly, contributors to the workflow should have different access rights on different parts of a document according to a document policy in order to protect both the confidentiality and integrity of documents. Thirdly, the fluidity of the workflow should be ensured by allowing exchange of documents by regular means, in addition to distributed and offline access, without violation of the security requirements.

Within the TVD framework, it is possible to enable fine-grained dissemination of information within a single workflow domain by integrating ERM features into the IT infrastructure. For that, a two-layered security policy enforcement can be employed to ensure document protection. The concept of TVDs is used

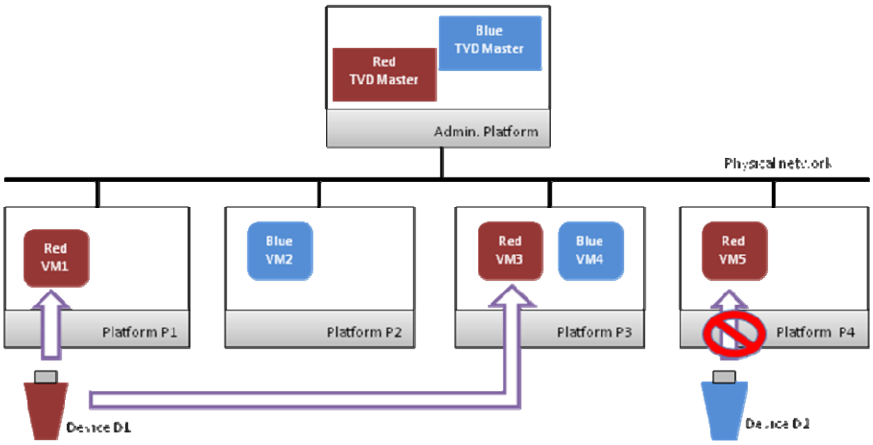

Figure 4: *Using MSDs in a system with two TVDs ("red" and "blue").*

to isolate workflows and tasks with different security requirements from each other, e.g., Internet access, intranet access, classified data access, and new document creation. Within a TVD, a trusted compartment on each of the contributing platforms – called *ERM controller* – enforces the document-level security policy which provides fine-grained dissemination of information.

While the isolation aspect of TVDs achieves the confidentiality requirement of classified data, some document workflows need a specific security concept supporting ERM features, such as encrypted storage and document-level policy enforcement. TVDs that are dedicated for document workflows and enforcement of document-level security policies are called ERM-TVDs. ERM-TVDs include compartments running a document policy enforcer (ERM controller). Moreover, the ERM security model benefits from the isolation capability of the underlying security kernel to restrict the access of the ERM application to network and storage resources through the trusted ERM controller. The same applies to the ERM controller itself, which can only access virtualized resources through trusted encryption modules which act as interfaces to encryption services.

One important advantage of this concept is that it allows to use existing operating systems and applications as ERM compartments, e.g., Open Office or MS Word, without relying on their security: if defined by the ERM-TVD policy, the underlying security kernel encrypts all persistent storage (hard disk, USB) and network traffic (VPN) using a TVD-specific cryptographic key. Moreover, this approach allows offline access to documents since the ERM-TVD policy is enforced based on the configuration of the security kernel which guarantees trustworthiness of the executed compartments. Since the ERM controller is part of the ERM-TVD, a violation of the document policy, e.g., due to a bug in the ERM controller, cannot violate the TVD policy. Therefore, the ERM controller can either be realized as a dedicated compartment running separately from the ERM application used to edit the document, or it can be an existing application running in the same compartment or a plug-in to the document rendering engine. Therefore, existing ERM controllers can also be used, depending on their compatibility with the required document security policy. This allows to realize fine-grained confidentiality and integrity requirements on parts of the document, based on a dedicated ERM controller for a specific document policy structure.

Protection of Mobile Storage Devices

As explained above, the security concept of TVDs can also be applied to secure the use of mobile storage devices. In this context, TVD infrastructures have to address two main objectives: On the one hand, they should be efficient enough to reduce the overhead of enforcing security policies; on the other hand, they have to be secure enough to reduce the efforts requested to users and consequently reducing the effects of human errors.

Figure 4 shows an example TVD-enabled infrastructure in which two different TVDs are deployed. Each physical platform runs one or more virtual machines belonging to one of the existing TVDs. Several USB memory sticks, variedly assigned to one of existing TVDs, are available to the users.

A typical usage example is as follows: The user Alice is working on the virtual machine VM1 and plugs in her USB stick D1 to the platform P1 to make a backup copy of her files. The TVD-enforcing components running on the platform P1 identify the plugged device, verify whether it has been assigned to the same TVD of VM1 and retrieve the cryptographic keys that are used to encrypt and decrypt data on it. If everything succeeds, the device is made available to VM1.

At this point, a further refinement to the device access control can be achieved on a per-VM basis. To this end, a set of rules that defines access privileges to each device assigned to the TVD (device access policy) has been added to the TVD policy. For each device, these rules state which operations and privileges (e.g., read, write) are granted to each virtual machine in the same TVD. Hence, the platform P1 allows VM1 to mount the device D1 under the constraints stated by the device access policy (read-only, read-write). Finally, if it is consistent with access privileges of VM1, the copy process of Alice's data can take place.

We recall that both device identification and key retrieval are performed automatically and transparently by the platform when the device is plugged in. The guest operating system of VM1 does not need any special software to open and access the device, and no additional operation from the user (e.g., further authentications besides login, or providing keys) is required to handle data contained on the device. Moreover, we stress that data encryption is mandatory, thus the user cannot choose to not encrypt data once the mobile storage device has been assigned to a TVD. Data stored on such a device can be accessed only by those virtual machines which joined the same TVD to which the data belongs to. Trying to access it through a virtual machine from a different TVD (or a computer outside the TVD) leads to a failure because the platform is not allowed to retrieve the corresponding encryption key.

Conclusion

Trusted Virtual Domains provide a secure IT infrastructure that enforces access control on data and network resources transparently and in a manageable fashion. The concept has been explored for several scenarios, ranging from data centers to workflow data protection in organizations and the usage of mobile storage devices. However, tools for TVD policy management are still not widely deployed. Besides concepts and prototypes from academia, products, such as Turaya, are developed and offered by companies. We expect IT systems based on TVDs will be a useful tool and investment for enterprises and organizations to protect their digital assets in shared IT infrastructures.

Bibliography

[1] BERGER, S., CACERES, R., PENDARAKIS, D. E., SAILER, R., VALDEZ, E., PEREZ, R., SCHILDHAUER, W., AND SRINIVASAN, D. TVDc: Managing security in the trusted virtual datacenter. Operating Systems Review 42, 1 (2008), 40–47.

[2] BUSSANI, A., GRIFFIN, J. L., JANSEN, B., JULISCH, K., KARJOTH, G., MARUYAMA, H., NAKAMURA, M., PEREZ, R., SCHUNTER, M., TANNER, A., DOORN, L. V., HERREWEGHEN, E. A. V., WAIDNER, M., AND YOSHIHAMA, S. Trusted Virtual Domains: Secure foundations for business and IT services. Tech. Rep. RC23792, IBM Research, 2005.

[3] CABUK, S., DALTON, C. I., RAMASAMY, H. V., AND SCHUNTER, M. Towards automated provisioning of secure virtualized networks. In Proceedings of the 2007 ACM Conference on Computer and Communications Security, CCS 2007, Alexandria, Virginia, USA, October 28-31, 2007 (2007), ACM, pp. 235–245.

[4] DISTRIBUTED MANAGEMENT TASK FORCE. ”Web-based Enterprise Management (WBEM)”. http://www.dmtf.org.

[5] GASMI, Y., SADEGHI, A.-R., STEWIN, P., UNGER, M., WINANDY, M., HUSSEIKI, R., AND STÜBLE, C. Flexible and secure enterprise rights management based on trusted virtual domains. In Proceedings of the 3rd ACM Workshop on Scalable Trusted Computing, STC 2008, Alexandria, VA, USA, October 31, 2008 (2008), ACM, pp. 71–80.

[6] GRIFFIN, J. L., JAEGER, T., PEREZ, R., SAILER, R., VAN DOORN, L., AND CACERES, R. Trusted Virtual Domains: Toward secure distributed services. In Proceedings of the 1st IEEE Workshop on Hot Topics in System Dependability (HotDep’05) (June 2005).

[7] KATSUNO, Y., KUDO, M., PEREZ, P., AND SAILER, R. Towards Multi-Layer Trusted Virtual Domains. In The 2nd Workshop on Advances in Trusted Computing (WATC 2006 Fall) (Tokyo, Japan, Nov. 2006), Japanese Ministry of Economy, Trade and Industry (METI).

[8] LÖHR, H., SADEGHI, A.-R., VISHIK, C., AND WINANDY, M. Trusted privacy domains – challenges for trusted computing in privacy-protecting information sharing. In Information Security Practice and Experience, 5th International Conference, ISPEC 2009 (2009), vol. 5451 of Lecture Notes in Computer Science, Springer, pp. 396–407.

[9] TRUSTED COMPUTING GROUP. TPM main specification, version 1.2 rev. 103, July 2007. https://www.trustedcomputinggroup.org.

[10] CATUOGNO, L., LÖHR, H., MANULIS, M., SADEGHI, A.-R., AND WINANDY, M. Transparent Mobile Storage Protection in Trusted Virtual Domains. In 23rd Large Installation System Administration Conference (LISA’09), USENIX Association, 2009

[11] EUROPEAN NETWORK AND INFORMATION SECURITY AGENCY (ENISA). Secure USB Flash Drives, June 2008. http://www.enisa.europa.eu /doc/pdf/Publications/SecureUSBdrives _180608.pdf.

[12] FABIAN, M. Endpoint security: managing USB-based removable devices with the advent of portable applications. In InfoSecCD’07: Proceedings of the 4th Annual Conference on Information Security Curriculum Development, ACM, pp. 1–5, 2007.

[13] BEAUTEMENT, A., COLES, R., J., IOANNIDIS, C., MONAHAN, B., PYM, D., SASSE, A., AND WONHAM, M. Modeling the human and technological costs and benefits of USB memory stick security. In Workshop on the Economics of Information Security (WISE’08), 2008.

[14] PARKIN, S. E., KASSAB, R. Y., AND VAN MOORSEL, A. P. A. The impact of unavailability on the effectiveness of enterprise information security technologies. In Service Availability, 5th International Service Availability Symposium, ISAS 2008, Tokyo, Japan, May 19-21, 2008, Proceedings, vol. 5017 of Lecture Notes in Computer Science, Springer, pp. 43–58, 2008.

[15] GOLDMAN, K., PEREZ R., SAILER, R. Linking remote attestation to secure tunnel endpoints. In Proceedings of the 1st ACM Workshop on Scalable Trusted Computing (STC’06), ACM Press, pp. 21–24, 2006.

[16] ARMKNECHT, F., GASMI, Y., SADEGHI, A.-R., STEWIN, P., UNGER, M., RAMUNNO, G., and VERNIZZI, D. An efficient implementation of trusted channels based on OpenSSL. In Proceedings of the 3rd ACM Workshop on Scalable Trusted Computing (STC’08), ACM, pp. 41–50, 2008.