# A gente descobre depois

João Gabriel Trombeta
João Paulo Taylor Ienczak Zanette
Ranieri Schroeder Althoff

8 de Abril de 2018

# Conteúdo

# Capítulo 1

# Máquinas Virtuais

## 1.1 Sobre Máquinas Virtuais

De maneira sucinta, máquinas virtuais (VM — *Virtual Machines*) são computadores sendo executados por outros computadores. Chama-se de **Guest** a máquina virtual em si, e de **Host** o *hardware* que oferece recursos para executar a VM.

## 1.2 Hypervisor

Também chamado de Monitor de Máquina Virtual (VMM — *Virtual Machine Monitor*), um **Hypervisor** é um componente (seja *hardware*, *software* ou *firmware*) responsável por criar e executar uma VM, sendo o *Host* o computador em que o Hypervisor é executado.

## 1.3 Falhas em Hypervisors

### 1.3.1 Xen Hypervisor

Uma falha de segurança detectada com relação a Hypervisors foi explorada no Xen Hypervisor (criado pelo Xen Project, composto por membros da The Linux Foundation), em que é possível chamar uma função arbitrária alterando a tabela de *Hypercalls* (semelhante a uma *vtable*). Uma *Hypercall* é *software trap* do Hypervisor para executar operações privilegiadas (como atualizar tabelas de página).

Para explorar a falha, primeiramente é necessário descobrir a localização da tabela de *Hypercalls*. Para isso, deve-se procurar pela assinatura da página. Porém, como a página não possui um formato tão previsível, é difícil de localizá-la (o que é feito pelo *checksum* do conteúdo da página). Em compensação, a tabela de argumentos dos *Hypercalls* possui um formato previsível, já que seu conteúdo — que é o número de argumentos de cada *Hypercall* — é fixo, e

portanto seu *checksum* também é previsível. Além disso, a tabela de argumentos sempre se encontra na página seguinte à tabela de *Hypercalls*, e portanto, ao encontrar uma, se tem a localização da outra. Aliado à possibilidade de leitura e escrita de código arbitrário, feito através de falhas nas regras de verificação de segurança de escrita em páginas do *Hypervision*, é possível então efetuar escape de máquina virtual (i.e. acessar recursos do *Host* que não pertencem à máquina virtual).

# Capítulo 2

# AMD Memory Encryption

## 2.1 Security Memory Encryption

## 2.2 Secure Encrypted Virtualization

## 2.3 Aplicação do SME e SEV

## 2.4

# Bibliografia

[1] H. Chen, X. Jia, and H. Li, "A brief introduction to iot gateway," in *IET International Conference on Communication Technology and Application (ICCTA 2011)*, pp. 610–613, Oct 2011.

[2] T. N. Stack, "Prototyping." https://thenewstack.io/tutorial-prototyping-a-sensor-node-and-iot-gateway-with-arduino-and-raspberry-pi-part-1/. [Online; acessado em 7 de abril de 2018].

[3] A. Botta, W. de Donato, V. Persico, and A. Pescapé, "Integration of cloud computing and internet of things: A survey," *Future Generation Computer Systems*, vol. 56, pp. 684 – 700, 2016.