Deep Attestation using Trusted Platform Module virtualization

Condrea Emil

Rezumat

Virtualizarea circuitului integrat Trusted Platform Module (TPM) este un pas înainte pentru securitatea cloud-ului. În această lucrare îmi propun să îmbunătățesc implementarea virtualizării TPM-urilor din hypervisor-ul Xen. Lucrarea va conține contribuții atât la nivel de kernel cât și la nivelul utilizator.

1. Propun o abordare diferită pentru atestarea în adâncime a autenticității unei mașini virtuale. Implementarea actuală nu funcționează pe majoritatea chip-urilor.
2. De asemenea nu există nici o implementare publică a unei comenzi de atestare. Voi veni cu contribuții și la biblioteca open source **trousers** care implementează și expune comenzile suportate de un TPM fizic.
3. Voi aduce contribuții și la suita de binare open source **tpm tools** prin implementarea unui program prin care se poate face atestarea în adâncime a unei mașini virtuale

   Toate proiectele vor fi implementate în C.

În această lucrare voi evidenția importanța TPM-ului în securitatea sistemelor de calcul. Este o lucrare de cercetare într-un domeniu cu documentație puțină. Proiectul aduce contribuții comunității open source. Sunt vizate: kernelul de Linux, Xen, emulatorul de TPM, *trousers* și *tpm tools*.

O primă problemă constatată este ca unele TPM-uri raportează greșit timeout-urile. Astfel se setează o valoare incorectă în driver după inițializare și modulul devine nefuncțional. Am investigat și am găsit o soluție pentru această problemă care este detaliată în capitolul 4.

Virtualizarea TPM-urilor în Xen conține 3 sub-entități [1]:

1. Domeniul pentru managerul de TPM-uri virtuale (*vtpmmgr*)
2. Domeniul pentru un TPM virtual (*vtpm*)
3. Emulatorul de TPM

Rolul managerului de TPM-uri este de a centraliza cererile către TPM-ul fizic și de a certifica în mod unic identitatea lor. El preia comenzi de la *vtpm*, le procesează dacă este cazul, apoi le trimite direct chip-ului.

TPM-ul are 5 pagini de memorie care pot fi accesate pentru comunicare. Ele pot fi accesate de diferite entități în funcție de nivelul de importanță a cererii. Pentru funcționalitate maximă *vtpmmgr* trebuie pornit cu nivelul de importanță 2. Problema care apare este că majoritatea TPM-urilor nu expun decât pagina de memorie cu nivelul de importanță 0. Astfel nu se poate face atestarea în adâncime a unei mașini virtuale.

Soluția pe care o propun este ca *vtpmmgr* sa folosească alte mecanisme de atestare care nu necesită accesul decât la pagina de memorie cu nivelul de importanță 0.

Rolul unui vTPM este de a primi cereri de la domeniul utilizator(*domU*), de a le procesa și apoi de a le redirecționa către *vtpmmgr*. El trebuie adaptat ca să știe să comunice cu noua versiune de *vtpmmgr*.

Pentru că nu toate comenzile cerute de domU necesită execuția pe TPM-ul fizic, *vtpm* folosește un emulator pentru a răspunde mai repede la comenzi. Emulatorul respectă standardul impus de Trusted Consulting Group(TCG) versiunea 1.2[2]. Comanda de atestare a *domU* necesită procesarea de *vtpmmgr*. Noua versiune propune o structură diferită a cererii.

Trousers este o bibliotecă open source care implementează TCG Software Stack (TSS) versiunea 1.2[3]. Biblioteca conține implementarea:

1. TCG Service Provider(TSP)
2. TCG Core Services (TCS)
3. Remote Procedure Calls (RPC)
4. Cryptographic Infrastructures.

Momentan nu există comenzi pentru atestarea *domU*. În această lucrare propun extinderea setului de comenzi și implementarea mai multor comenzi printre care și *DeepQuote*. Modificările trebuie făcute atât la layerul TSP, RPC cât și la TCS.

Tpm tools este o suită de binare care integrează *trousers* pentru a expune o funcționalitate utilizatorului final. În lucrare propun o extindere a setului de binare (executabile) pentru a expune funcționalitatea de atestare în adâncime. Noua versiune va folosi și comanda *DeepQuote*.

Implementarea proiectului este terminată. Am trimis modificările sub formă de patch-uri pe lista de discuții oficială a proiectului Xen și am primit feedback foarte bun.

Bibliografie

1: Matthew Fioravante (JHUAPL), Daniel De Graaf (NSA), Virtual TPMs architecture, http://xenbits.xen.org/docs/unstable/misc/vtpm.txt

2: TCG, TPM Main Part1 Design Principles, 2011, http://www.trustedcomputinggroup.org/files/static_page_files/72C26AB5-1A4B-B294-D002BC0B8C062FF6/TPM%20Main-Part%201%20Design%20Principles_v1.2_rev116_01032011.pdf

3: TCG, TCG Software Stack (TSS) Specification Version 1.2, 2007, http://www.trustedcomputinggroup.org/files/resource_files/6479CD77-1D09-3519-AD89EAD1BC8C97F0/TSS_1_2_Errata_A-final.pdf