PROJEKT

Multitenantné operačné centrum kybernetickej bezpečnosti riešené ako otvorená cloudová služba s prvkami strojového učenia

Previazané s balíkom KPB6 – Publikačné výstupy

Financované Európskou úniou
NextGenerationEU

PLÁN OBNOVY

ÚRAD VLÁDY
SLOVENSKEJ REPUBLIKY

Tvorba SECaaS služby v privátnom cloude OpenStack

Obsah

Zoznam obrázkov **10**
Zoznam tabuliek **12**
Zoznam skratiek **13**
Úvod **16**
1Súčasný stav riešenia problematiky **18**
1.1Štandardy súvisiace s CC 18
1.1.1Štandard ITU-T Y.3500 [4] 18
1.1.2Štandard ITU-T Y.3501 [6] 22
1.1.3Štandard ITU-T Y.3502 [7] 24
1.1.4Štandard ITU-T X.1602 25
1.2Existujúce riešenia SECaaS služieb 31
1.2.1Prehľad poskytovateľov SECaaS 32
1.2.2Zhodnotenie analýzy súčasných riešení 35
1.3Openstack 35
1.3.1Využitie OpenStack CC nielen v kritickej infraštruktúre 36
1.3.2Komponenty CC OpenStack 38
1.3.3Placement 42
1.3.4Popis uzlov v OpenStacku 42
1.3.5DevStack 44
1.3.6Komponenty zabezpečenia implementované v OpenStack 45
2Ciele práce **54**
3Metodika práce a metódy skúmania **55**
4Výsledky Práce **58**
4.1.1Analyzovanie možností nasadenia SECaaS na OpenStack 58
4.2Príprava testovacieho prostredia 66
4.2.1Zabezpečenie testovacieho prostredia 66
4.2.2Inštalácia cloudu pomcou DevStack 67
4.2.3Sieťová konfigurácia cloudu 69
4.2.4Testovacia topológia 70
4.2.5Vzdialený prístup k zariadeniam 71
4.3Implementácia cloudovej IDS/IPS služby 75
4.3.1Scenár 1 – Sieťový IDS cez TAPaaS 76
4.3.2Scenár 2 – Transparentný IPS 83
4.4Implementácia služby pre posudzovanie zraniteľností cloudu 88
4.4.1Greenbone Vulnerability Management [35] 88

4.4.2Inštalácia GVM 89
4.4.3Úprava zabezpečovacieho skriptu 89
4.4.4Testovanie GVM as a Service (GVMaaS) a vyhodnotenie 91
4.5Diskusia k výsledkom 94
Záver 96
Zoznam použitej literatúry 98
Príloha A: Obsah CD 103
Príloha B: Skript pre zabezpečenie cloudu 104
Príloha C: Súbor „local.conf" obsahujúci konfiguráciu DevStack 109
Príloha D: Konzolový výstup po úspešnej inštalácií DevStack 112

Zoznam obrázkov

Obrázok 1 – Pohľady v CCRA, spracované podľa ITU-T Y.3502 [7] 25
Obrázok 2 – Konceptuálny model pre SaaS aplikačné prostredie [8] 26
Obrázok 3 – Diagram úrovní zrelosti SaaS aplikácie [8] 27
Obrázok 4 – Prispôsobené SaaS aplikácie [8] 27
Obrázok 5 – Konfigurovateľné SaaS aplikácie [8] 28
Obrázok 6 – SaaS aplikácie pre viacerých používateľov [8] 29
Obrázok 7 – Škálovateľné SaaS aplikácie [8] 30
Obrázok 8 – Vzťahy pre SaaS aplikačné prostredie [8] 30
Obrázok 9 – Architektonický rámec Hytrust CloudSPF [11] 33
Obrázok 10 – Odporúčané rozloženie cloudu z pohľadu uzlov [19] 43
Obrázok 11 – Rozloženie privátneho cloudu katedry informačných sietí FRI UNIZA 44
Obrázok 12 – Porovnanie FWaaS a bezpečnostných skupín 47
Obrázok 13 – Schéma nasadenia TAPaaS [21] 49
Obrázok 14 – Objektový model TAPaaS [22] 50
Obrázok 15 – Model overovania pomocou SSH certifikátov 52
Obrázok 16 – Proces riešenia problému 56
Obrázok 17 – Architektúra NIDSaaS [26] 61
Obrázok 18 – Postup pri práci s NIDSaaS [26] 62
Obrázok 19 – Sieťový model NIDSaaS [26] 64
Obrázok 20 – Infraštruktúra pre testovanie zraniteľností cloudu [27] 65
Obrázok 21 – Inštalácia a príprava DevStack-u 68
Obrázok 22 – Možnosti prepnutia verzií inštalovaného cloudu 68
Obrázok 23 – Prepnutie Iptables do legacy režimu 69
Obrázok 24 – Príkazy pre otvorenie cloudových inštancií pre externé prostredie 70
Obrázok 25 – Testovaná virtuálna infraštruktúra v cloude 71
Obrázok 26 – Diagram prípadov použitia pri pristupovaní ku cloudu 73
Obrázok 27 – Nastavenie SSH agenta 74
Obrázok 28 – Logovacie výpisy týkajúce sa inicializačných dát pri bootovaní zariadení 74
Obrázok 29 – Overenie prístupu vo vytvorenej virtuálnej infraštruktúre 75
Obrázok 30 – Implementácia IDS služby v infraštruktúre 76
Obrázok 31 – Vytvorenie TAP služby 77
Obrázok 32 – Vytvorenie TAP flow 78

Obrázok 33 – Vypísanie vytvorených TAPaaS objektov78
Obrázok 34 – Overenie zrkadlenia79
Obrázok 35 – Overenie inštalácie Suricaty na Jump-in VM80
Obrázok 36 – Upravenie konfigurácie Suricaty pre scenár 181
Obrázok 37 – Generovanie IP prevádzky útočníkom (VM2) pre test pridaných IDS pravidiel81
Obrázok 38 – Úspešne identifikovanie útočnej prevádzky v logoch Suricaty82
Obrázok 39 – Diagram prípadov použitia cloudového IDS83
Obrázok 40 – Implementácia IPS služby v infraštruktúre84
Obrázok 41 – Porovnanie systémových záťaží86
Obrázok 42 – Overenie počtu zablokovaných paketov87
Obrázok 43 – Logy detekcie a zahodenie škodlivej IP prevádzky na Router VM87
Obrázok 44 – Diagram prípadov použitia cloudového IPS88
Obrázok 45 – Sprístupnenie GVM v infraštruktúre do internetu90
Obrázok 46 – Nastavenie presmerovania portov pre pristupovanie ku GVM mimo cloudu90
Obrázok 47 – Úspešný mimo-cloudový prístup ku webovému rozhraniu (GSA) pre GVM91
Obrázok 48 – Skenované zariadenia nástrojom GVM92
Obrázok 49 – Bezpečnostné riziká generované GVM93

Zoznam tabuliek

Tabuľka 1 – Kategórie CC služieb pridružené k typom CC [4] 21
Tabuľka 2 – Bezpečnostné požiadavky CSP a CSN v SaaS 31
Tabuľka 3– Analýza poskytovateľov SECaaS a priradenie kategóriám SECaaS 35
Tabuľka 4 – Zoznam identifikovaných portov v infraštruktúre 94

Zoznam skratiek

API	Application Programming Interface
APT	Advanced Package Tool
ASA	Adaptive Security Appliance
AWS	Amazon Web Services
BC/DR	Business continuity and disaster recovery
CA	Certifikačná Autorita
CC	Cloud Computing
CCRA	Cloud Computing Reference Architecture
CLI	Command Line Interface
CSC	Cloud Sevice Customer
CSN	Cloud Service Partner
CSP	Cloud Service Provider
DaaS	Desktop as a Service
DoS	Denial of Service
ET	Emerging Threats
FWaaS	Firewall as a Service
GDPR	General Data Protection Regulation
GIT	GNU Interactive Tools
GNOME	GNU Network Object Model Environment
GNU	GNU's Not Unix
GSA	Greenbone Service Assistant
GVM	Greenbone Vulnerability management
GVMaaS	Greenbone Vulnerability management as a Service
HOT	Heat Orchestration Template
IaaS	Infrastructure as a Service
IaC	Infrastructure as a Code
ICMP	Internet Control Message Protocol

IDS	Intrusion Detection System
IKT	Informačné a komunikačné technológie
IPS	Intrusion Prevention System
ITU	The International Telecommunication Union
JSON	JavaScript Object Notation
KRL	Key Revocation List
LTS	Long Term Support
MITM	Man in the middle
NAT	Network Address Translation
NFQ	Net Filter Queue
NFW	Network Function Virtualization
NGFW	Next-Generation Firewall
NIDS	Network Intrusion Detection System
NIDSaaS	Network Intrusion Detection System as a Service
OISF	Open Information Security Foundation
OVS	Open vSwitch
PaaS	Platform as a Service
PAT	Port Address Translation
PDF	Portable Document Format
QoD	Quality of Detection
RDP	Remote Desktop Protocol
REST	Representational state transfer
RPC	Remote Procedure Call
SaaS	Software as a Service
SAP	Service Access Point
SDLC	System Development Life Cycle
SDN	Software Defined Networking
SECaaS	Security as a Service
SPAN	Switch Port Analyzer

SSH	Secure Shell
TAP	Test Access Point
TAPaaS	Test Access Point as a Service
TCP	Transmission Control Protocol
TLS	Transport Layer Security
TOFU	Trust On First Use
UDP	User Datagram Protocol
UFW	Uncomplicated Firewall
VAaaS	Vulnerability Assessment as a Service
VIO	VMware Integrated OpenStack
VLAN	Virtual Local Area Network
VM	Virtual Machine
VNC	Virtual Network Computing
VoIP	Voice over Internet Protocol
VPN	Virtual Private Network
VXLAN	Virtual eXtensible Local Area Network
YAML	YAML Ain't Markup Language

1 Úvod

Cloud computing (CC) predstavuje dynamicky sa rozvíjajúcu oblasť informačných technológií s rastúcim portfóliom služieb, ktoré nachádzajú uplatnenie v akademickom, komerčnom aj verejnom sektore. Napriek mnohým výhodám, ktoré CC prináša, zostáva zabezpečenie cloudovej infraštruktúry, dát a služieb kľúčovou výzvou. Kybernetické hrozby môžu ovplyvniť dostupnosť, dôvernosť aj integritu dát, čo si vyžaduje nové prístupy k ich ochrane. Jedným z perspektívnych riešení je koncept bezpečnosti ako služby – Security as a Service (SECaaS), ktorý predstavuje moderný spôsob poskytovania bezpečnostných funkcií prostredníctvom cloudových technológií.

SECaaS umožňuje zákazníkom využívať centralizované bezpečnostné služby, pričom zodpovednosť za ich implementáciu a správu preberá poskytovateľ. Takýto model prináša viacero výhod – od možnosti monitorovania v reálnom čase, cez automatizáciu bezpečnostných procesov, až po efektívnu reakciu na incidenty a tvorbu štatistických prehľadov o hrozbách. Na druhej strane však SECaaS prináša aj určité riziká, ako je potenciálne ohrozenie viacerých klientov pri útoku na infraštruktúru poskytovateľa alebo pri nedostatočne oddelenej architektúre služby.

Cieľom nášho výskumu bolo vypracovanie metodiky pre implementáciu a poskytovanie vybraných SECaaS služieb v prostredí privátneho cloudu založeného na platforme OpenStack. Výskum reflektuje potrebu dostupného a flexibilného riešenia bezpečnostných služieb najmä v akademickom prostredí, kde je možné experimentovať a overovať nové prístupy. Praktická časť výskumu bola realizovaná formou proof of concept na Katedre informačných sietí Fakulty riadenia a informatiky Žilinskej univerzity v Žiline na Slovensku. Teoretická časť sa sústredila na analýzu súčasného stavu v oblasti SECaaS, štandardov cloudovej bezpečnosti a možností ich aplikácie v rámci OpenStack. Nasledujúce kapitoly článku prezentujú navrhnutú metodiku, realizáciu a výsledky testovania v pilotnom prostredí.

2 Súvisiace práce

Network Intrusion Detection System as a Service v OpenStacku

Sieťový IDS je podľa [27] nevyhnutným nástrojom na detekovanie kybernetických útokov na cloudy. Jednou z možných riešení pre používateľov cloudových služieb je zabezpečenie ich virtuálnych strojov bežiacich na cloude implementovaním IDS pri vstupe do ich sietí, kde budú IDS služby spustené ako inštancie vnútri virtuálnych strojov. Tento prístup však predstavuje značnú virtuálnu záťaž a je časovo náročný vzhľadom na tvorbu IDS služby pre každého používateľa cloudu. Práca [27] navrhuje riešenie, ktoré poskytuje dostupné a rýchle nasadenie a terminovanie IDS služieb, zatiaľ čo systémová záťaž pre cloudových poskytovateľov zostane nízka. Autori práce navrhli a implementovali prototyp Network Intrusion Detection System as a Service (NIDSaaS), ktorý sa následne vyhodnotil na testovacom prostredí OpenStack.

Cloudové systémy určené pre privátne nasadenie, napr. Openstack, Eucalyptus, alebo OpenNebula poskytujú viacero služieb na splnenie svojich potrieb a potrieb ich zákazníkov. Tieto riešenia však často nemajú funkciu pre detekovanie škodlivej prevádzky pre ochranu inštancií zákazníkov proti rôznym kybernetickým útokom. Často býva prípadom, že útoky neprichádzajú z externého prostredia, ale vznikajú priamo v internom prostredí organizácie. Jeden používateľ cloudových služieb, ktorý chce zaútočiť na iných používateľov má prístup k rôznym nástrojom, medzi ktoré patria napríklad port skenery, nástroje pre analyzovanie IP prevádzky, programy pre vykonávanie spoofing útokov, a pod. Na zabránenie útokov jednak z externého, ale aj z interného prostredia, by mali cloudový administrátori implementovať strategicky umiestnené IDS systémy pre odhalenie škodlivej prevádzky v cloude alebo pre vykonávanie aktívneho blokovania útokov nasadiť IPS s požadovanými signatúrami.

Súčasne prístupy pre monitorovanie prevádzky používateľov a detekcie vniknutí do ich siete používajú softvérovo definované siete a virtualizačné technológie. Vo fyzickom prostredí, Switched Port Analyzer (SPAN) na fyzickom prepínači duplikuje pakety smerom na zariadenie, na ktorom je nasadený IDS. V prípade virtuálnych prostredí dokáže byť funkcionalita SPAN implementovaná pomocou virtuálneho prepínacieho zariadenia (pre OpenStack buď Open vSwitch OVS, alebo Linux Bridge). Takýto nástroj umožnil vytvorenie virtuálnych IDS služieb bežiacich v cloude. Jedným problémom pri monitorovaní prevádzky v sieti zákazníka je situácia, kedy má daný zákazník sieť zariadení rozloženú cez viaceré cloudové prostredia. Riešením pre tento problém je OpenStack služba

s názvom TAP as a Service. TAP as a Service (TAPaaS , alebo TaaS) – TAP = Test Access Point, je cloudová sieťová služba, ktorá umožňuje kopírovať (zrkadliť) sieťovú prevádzku z jednej virtuálnej siete (napríklad portu alebo inštancie) do inej – zvyčajne pre účely monitorovania, analýzy, bezpečnosti alebo ladenia prevádzky. Pomocou TAPaaS dokážeme monitorovať prevádzku naprieč viacerými cloudovými servermi. Používatelia cloudov môžu mať spustenú IDS službu vytvorením virtuálnej inštancie zariadenia v cloude, na ktorom spustia IDS nástroj a ktoré bude cieľom pre monitorovanú sieťovú prevádzku.

Práca [27] navrhuje model IDS služby pre OpenStack s názvom NIDSaaS (Network Intrusion detection system), ktorá poskytuje IDS služby na požiadanie zákazníkom cloudu. Služba je navrhnutá tak, aby dokázala dynamicky vytvoriť, prípadne vymazať, IDS služby v OpenStack cloude, a aby bola schopná efektívne aktualizovať súbory pravidiel týkajúcich sa IDS. Takéto riešenie poskytuje lepšie spravovanie IDS služieb a zároveň redukuje virtuálizačnú záťaž.

Prototyp služby NIDSaaS navrhnutý v [27] pozostáva z klienta pre spravovanie služby, pluginu služby, agenta pluginu služby a Snort IDS. Služba pracuje na OpenStack riadiacom uzle, na sieťových a výpočtových uzloch.

Nulová dôvera (Zero-Trust Architecture)

Nulová dôvera sa stala základom cloudovej bezpečnosti v roku 2025, najmä kvôli nárastu vzdialenej práce a decentralizovaných IT prostredí, ktoré odhalili nové zraniteľnosti. Výskum naznačuje, že 87 % organizácií sa zameriava na overovanie prístupu na základe identity, zariadenia a lokality, nasledovaním princípu najmenších práv (least privilege). Tento trend je poháňaný nárastom kybernetických útokov o 47 % a výzvami v oblasti kybernetickej hygieny koncových bodov u 70 % organizácií (Wipro Cybersecurity Report [https://www.wipro.com/content/dam/nexus/en/service-lines/applications/latest-thinking/state-of-cybersecurity-report-2020.pdf]). Nulová dôvera znižuje riziko neoprávneného prístupu, čo je kľúčové pre SECaaS v multi-cloudových a hybridných prostrediach.

AI a strojové učenie (ML)

AI a ML sú považované za nevyhnutné pre cloudové stratégie, pričom 90 % organizácií verí, že sú potrebné pre efektívnu bezpečnosť. Výskum ukazuje, že 45 %

organizácií verí, že AI prevyšuje ľudských analytikov, a 32 % plánuje rozsiahle investície do AI v priebehu 12–18 mesiacov. Očakáva sa nedostatok 1,8 milióna pracovníkov v oblasti kybernetickej bezpečnosti, čo zvyšuje závislosť na automatizovaných nástrojoch. AI sa používa na prediktívnu analýzu dát, detekciu anomálií a automatizáciu odpovedí na hrozby, čo je neočakávané spojenie technológií pre laikov, ktorí by mohli predpokladať, že bezpečnostné služby sú čisto manuálne alebo založené na pravidlách (Cloud Security Trends 2025 [https://www.cloudpanel.io/blog/cloud-security-trends/]).

2.4 Štandardy súvisiace s Cloud computing

Komentár od [STUD - Pe1]: potrebujeme mať v článku zmienku o štandardoch?

Štandard ITU-T Y.3500 [5]

Štandard definuje pojmy a poskytuje všeobecný prehľad pre oblasť CC.

V [5] je uvedené, že „CC umožňuje prístup k škálovateľnému a elastickému súboru zdieľaných fyzických prvkov alebo virtuálnych zdrojov s poskytovaním samoobsluhy s možnosťou administrácie.“ Medzi základné vlastnosti CC patrí sieťový prístup, čo znamená, že CC je prístupný odkiaľkoľvek zo siete. Ďalšou vlastnosťou je Spoločné využívanie, kedy CC je zdieľaný medzi viacerých používateľov, ktorí si navzájom nezasahujú do svojej časti cloudu. Charakteristickou črtou je priradenie zdrojov na požiadanie, čo znamná, že množstvo použitých zdrojov je uživateľovi dynamocky pridelované na základe jeho potrieb. Existuje funkcia, vďaka ktorej je možné upravovať pridelené zdroje pri zmene potrieb čo je škálovateľnosť a elasticita. Pre CC je charakteristické združovanie fyzických zdrojov tak aby slúžili samostatne používateľom.

Štandard ITU-T Y.3501

Štandard ITU-T Y.3501 definuje všeobecné požiadavky pre cloud computing (CC) služby, ktoré zahŕňajú manažment životného cyklu služieb, dodržiavanie legislatívy (najmä v oblasti ochrany osobných údajov), bezpečnosť, efektívne zavádzanie služieb, interoperabilitu, prenosnosť, dostupnosť z rôznych zariadení, spoľahlivosť a kvalitu služieb. Tieto požiadavky sú spoločné pre všetky typy cloudových služieb a ich dodržiavanie je dôležité najmä pri návrhu nových riešení.

Pre infraštruktúrne služby (IaaS) štandard požaduje zabezpečenie abstrakcie a kontroly zdrojov, ako aj ich efektívne poskytovanie a monitorovanie. Dôraz sa kladie na spracovanie, ukladanie a sieťové zdroje ako základné stavebné prvky cloudovej infraštruktúry.

Štandard zároveň špecifikuje požiadavky pre poskytovanie dôveryhodných cloudových služieb. Tie zahŕňajú kontrolu nad službami v súlade s normami, implementáciu mechanizmov na správu a lokalizáciu dát, ochranu súkromia a bezpečné vymazanie dát. Dôveryhodnosť služieb je podporená aj odolnosťou voči incidentom, zabezpečením prostredníctvom kryptografie, detekcie útokov, dostupnosti pre používateľov a schopnosťou auditovania. Neoddeliteľnou súčasťou je aj dodržiavanie servisných dohôd so zákazníkmi.

Štandard ITU-T Y.3502 [8]

Štandard ITU-T Y.3502, obsahuje referenčnú architektúru CC (CCRA – Cloud Computing Reference Architecture). CCRA poskytuje architektonický framework, ktorý efektívne opisuje úlohy, činnosti a aspekty pre CC. CCRA sa zaoberá:

- Opisom komunity zúčastnených strán v oblasti CC.
- Opisom základných charakteristík CC systémov.
- Špecifikovaním základných činností v CC, funkčných komponentov CC a popisovaniu ich vzájomných vzťahov.
- Identifikovaním princípov, ktorými sa riadi návrh a vývoj CCRA.

CCRA môže byť použitý ako základný referenčný bod pre štandardizáciu CC. Poskytuje okrem toho aj dodatočné informácie ohľadom základnej koncepcie a princípov pri tvorbe cloudových služieb.

Štandard ITU-T X.1602

Štandard [9] sa týka vývoja cloudových aplikácií, konkrétne požiadaviek kladených na bezpečnosť služby pre zabezpečenie konzistentného a bezpečného prostredia pre SaaS aplikácie (Software as a Service). Tieto navrhované požiadavky sú určené pre poskytovateľov CC a partnerov cloudových služieb, pretože potrebujú, aby SaaS aplikácie spĺňali požiadavky na bezpečnosť.

Konceptuálny model pre SaaS aplikačné prostredie

Základné funkcie cloudovej aplikácie zoskupujeme do služieb a poskytujeme konzistentný zabezpečený prístup pomocou prístupového bodu služby (Service Access Point – SAP). Takto by bolo možné zabezpečiť SaaS aplikačné prostredie tak že:

- IaaS poskytuje výpočtové, úložné a sieťové služby.
- PaaS poskytuje platformové služby.
- DaaS poskytuje desktopové služby.

Komentár od [STUD - Pe2]: desktopové služby sú aké?

Bezpečnostné požiadavky pre SaaS aplikačné prostredie

V aplikačnom prostredí majú poskytovateľ (Cloud Service Provider – CSP), zákazník (Cloud Service Customer – CSC) a partner (Cloud Service Partner – CSN) CC vlastné role pri vykonávaní rôznych funkcií. Majú povinnosť spĺňať bezpečnostné požiadavky, keďže len oni prichádzajú do styku s vývojom aplikačného prostredia. Samotní zákazníci konečnú aplikáciu len využívajú.

2.5 Existujúce riešenia SECaaS služieb

S rozvojom cloudových technológií vznikajú nové služby, ktoré umožňujú flexibilne reagovať na potreby zákazníkov. Jednou z nich je SECaaS (Security as a Service) – služba poskytovaná externou firmou na zabezpečenie cloudovej infraštruktúry zákazníka. Využíva bezpečnostný softvér nasadený v cloude, prostredníctvom ktorého poskytovateľ riadi bezpečnosť cloudových sietí.

SECaaS riešenia sa líšia podľa rozsahu služieb, spôsobu doručovania či cieľovej skupiny (firmy vs. súkromní používatelia). Zákazníci za tieto služby zvyčajne platia mesačný poplatok, ktorý zahŕňa prístup k bezpečnostným nástrojom a ochranu aplikácií, dát a cloudových procesov.

Organizácia Cloud Security Alliance (CSA) stanovila kategórie SECaaS služieb, medzi ktoré patrí správa identity a prístupu, prevencia straty dát, webová ochrana, šifrovanie, detekcia zraniteľností, manažment bezpečnostných incidentov, sieťová ochrana, aplikačná bezpečnosť, BC/DR, monitorovanie prevádzky, ochrana e-mailov a spam filtrácia. Mnohé SECaaS riešenia kombinujú viacero z týchto funkcií.

Výsledok analýzy súčasných riešení

Zistili sme, že na trhu existuje niekoľko firiem, ktoré poskytujú služby zabezpečenia cez cloud. Každý poskytovateľ sa však sústreďuje na rôzne aspekty zabezpečenia svojich zákazníkov. Služby SECaaS sa líšia napríklad v charaktere poskytovaných služieb, v spôsobe doručovania služby, zamerania na firmy alebo privátnych používateľov.

CSA, nezisková organizácia zameraná na rozšírenie osvedčených postupov pre zaručenie bezpečnosti v CC, vytýčila niekoľko kategórií SECaaS-u. Jednotlivé riešenia od poskytovateľov služby môžu zahŕňať v sebe len jednu kategóriu SECaaS-u, alebo priamo niekoľko.

Komentár od [STUD - Pe3]: vyhodiť prehľad poskytovateľov, -jedine, že by si miesto v článku chceli zaplatiť

Komentár od [STUD - Pe4]: prepracovať

2.6 OpenStack a SECaaS služby

Platformu OpenStack, na ktorej sme realizovali výskum SECaaS, charakterizujeme ako otvorený štandard CC systému, zameraného prevažne na prevádzkovanie IaaS v privátnych alebo verejných cloudoch.

Komentár od [STUD - Pe5]: zovšeobecniť

Za zrodom tejto cloudovej platformy stojí americká vesmírna agentúra NASA a spoločnosť Rackspace Technology. V roku 2010 spojili sily s cieľom vytvoriť cloudovú platformu založenú na open source, ktorú bude možné ľahko používať, implementovať, bude mať široký rozsah možných použití a bude spĺňať potreby používateľov a poskytovateľov verejných a privátnych cloudov [17].

Koncový používatelia systému majú prístup k prideleným výpočtovým zdrojom, ktoré spravuje administrátor cloudu. Systém pozostáva z niekoľko navzájom prepojených modulov umožňujúcim vykonávať správu a prideľovanie zdieľaného výpočtového výkonu [15]. Používatelia majú možnosť spravovať tieto zdroje pomocou webového rozhrania, CLI (Command Line Interface) nástrojov alebo REST (Representational state transfer) API služby. Platformu je možné spustiť len na linuxových operačných systémoch. Aktuálna verzia OpenStack v čase výskumu je 2025.1 s kódovým označením “Epoxy“.

Medzi výhody OpenStacku považujeme vysokú úroveň zabezpečenia a spoľahlivosti dát. Úložisko a výpočtový výkon sa dá jednoducho škálovať v prípade, že sú prostriedky nepostačujúce pre momentálne používanie systému. Prístup ku cloudovým službám nie je limitovaný lokáciou ani používaným zariadením. OpenStack je využívaný v rôznych odvetviach firmami rozličných veľkostí a za jeho úspechom stojí silná komunita používateľov a vývojárov.

Privátny cloud katedry informačných sietí FRI UNIZA pozostáva z troch serverov. Na prvom je spustený riadiaci a sieťový uzol. Druhý výpočtový uzol poskytuje výpočtový výkon cez VM a posledný uzol slúži pre spúšťanie kontajnerovaných služieb. Jednotlivé servery sú prepojené pomocou fyzického prepínacieho zariadenia Cisco Catalyst 4948. Každý server disponuje dvomi fyzickými sieťovými rozhraniami. Prvé fyzické rozhranie je zaradené do manažmentovej Virtual Local Area Network (VLAN) pre prenos riadiacej IP prevádzky medzi komponentami architektúry. Druhé fyzické rozhranie je ďalej rozdelené na dve virtuálne rozhrania. Prvé virtuálne rozhranie prenáša neznačkovanú IP prevádzku z externej siete a druhé za pomoci technológie VXLAN tuneluje prevádzku zákazníkov CC medzi jednotlivými fyzickými servermi. Medzi internetom a prepínačom sa nachádza Cisco Adaptive Security Appliance (ASA) 5520 pre vykonávanie firewallových, VPN, NAT (Network Address Translation) a iných funkcií.

V systéme OpenStack existujú služby pre poskytovanie zabezpečenia v cloude. Tieto služby môžu byť priamo integrované v komponentoch OpenStacku alebo sú implementované v podobe zásuvných modulov pre jednotlivé komponenty. Uvedieme si niekoľko z týchto služieb, ktoré môžu byť využité pri poskytovaní SECaaS.

Bezpečnostné skupiny (Security Groups)

Bezpečnostné skupiny v OpenStacku sú pomenovaným súborom prístupových pravidiel pre pakety, cez ktoré je možné limitovať sieťové toky z alebo do virtuálnych inštancií.

Firewall as a service (FWaaS) Neutron plugin

FWaaS v OpenStacku je zásuvný modul pre Neutron službu na vykonávanie paketového filtrovania prevádzky pri vstupe do sietí v cloude.

Test Access Point as a service (TAPaaS) Neutron plugin

Test Access Point as a Service [22], TAPaaS je rozvinutá sieťová služba spustená v OpenStacku rozširujúca jeho funkcionalitu o možnosť zrkadlenia IP prevádzky. Funguje ako rozšírenie Neutron komponentu. TAPaaS berie do úvahy hranice používateľských sietí a je možné ju využiť aj v cloude zloženom z viacerých sieťových a výpočtových uzlov. Slúži ako dôležitý prvok infraštruktúry, ktorý je možné využiť na sprostredkovanie dát o sieťovej

prevádzke buď pre analýzu, odladenie interných cloudových procesov alebo pre zakomponovanie bezpečnostných prvkov.

Virtual Private Network as a Service Neutron plugin

Ďalší zásuvný modul pre Neutron VPNaaS pridáva do OpenStack funkcionalitu z oblasti virtuálnych privátnych sietí. Účelom služby je podpora viacerých tunelovacích a bezpečnostných protokolov s podporou statického aj dynamického smerovania, no v súčasnej dobe poskytuje základnú implementáciu IPsec VPN na základe open source nástrojov so statickým smerovaním.

Tatu – Secure Shell as a service (SSHaaS)

Tatu je dodatočná služba pre OpenStack pre správu používateľských a serverových SSH certifikátov. Služba navrhuje riešenie problému správy vzdialených prístupov do virtuálnych inštancií v cloude využitím SSH certifikátov. SSH certifikáty boli predstavené vo verzii OpenSSH 5.4. Princíp fungovania SSH certifikátov je založený na myšlienke, že klienti a servery si nemajú navzájom dôverovať, ale majú si overovať informácie o druhej strane voči certifikačnej autorite (Certification Authority – CA).

3 Metodika práce a metódy skúmania

Cieľom výskumu bolo preskúmať možnosti nasadenia SECaaS služieb v privátnom cloude OpenStack a následne otestovanie a vyhodnotenie jednotlivých riešení.

K dosiahnutiu cieľa bol postup práce rozdelený do niekoľkých parciálnych cieľov:

- Analyzovanie aktuálneho portfólia poskytovaných SECaaS služieb.
- Analyzovanie dostupných štandardov týkajúcich sa služieb CC.
- Identifikovanie bezpečnostných nástrojov ponúkaných v rámci SECaaS.
- Spomedzi identifikovaných nástrojov si vybrať podľa zadefinovaných kritérií jeden, prípadne viaceré nástroje, do implementačnej časti pre nasadenie SECaaS služby v privátnom cloude OpenStack.
- Spracovanie metodiky vývoja služby SECaaS podľa štandardov pre CC.
- Vytvorenie SECaaS služby v katedrovom privátnom cloude.
- Otestovanie funkčnosti vytvorenej služby a vyhodnotiť jej efektivitu a ďalšie využitie.

Pri analýze sme zbierali teoretické znalosti tykajúce sa CC a vývoju CC služieb. Ďalej sme sa v kapitole Existujúce riešenia SECaaS služieb pozreli, aké súčasné riešenia existujú a na akú formu zabezpečenia sa zameriavajú poskytovatelia SECaaS služieb.

Vedomosti nadobudnuté počas prvých dvoch fáz sme zužitkovali pri návrhu SECaaS. V OpenStacku je možné realizovať zabezpečenie cez prvky implementované v systéme alebo prostredníctvom bezpečnostných služieb bežiacich na VM. Pre SECaaS je potrebné si spomedzi identifikovaných bezpečnostných nástrojov ponúkaných v OpenStacku vybrať konkrétne a využiť ich pri realizácii služby. **Odoberateľ IaaS/PaaS služieb by mohol zdieľané cloudové výpočtové zdroje použiť na iné účely, než na tie, kvôli ktorým mu boli pridelené. Preto implementujeme službu skenovania cloudového prostredia, ktorá dokáže identifikovať služby bežiace v cloude a taktiež odhalí potencionálne softvérové hrozby konkrétnych verzií softvérov na spustených VM.**

Komentár od [STUD - Pe6]: dôležité

Fáza realizácie začala vytvorením izolovaného cloudového prostredia pre testovanie vyvíjaných služieb. Toto prostredie bolo spustené na virtuálnom stroji poskytnutom katedrou KIS. Pri inštalácii cloudu bol použitý nástroj DevStack. Skúmali sme jednotlivé možnosti nasadenia navrhnutých bezpečnostných služieb na základe nástrojov zakomponovaných v OpenStacku popísaných vo fáze analýzy a prác z databázy IEEExplore zaoberajúcimi sa implementovaním bezpečnostných služieb v CC OpenStack. Služby boli otestované vo virtuálnej infraštruktúre zariadení, ktorá bola vytvorená na základe HOT skriptov. Museli sme zaviesť mechanizmus pre vzdialený prístup k zariadeniam nachádzajúcich sa v infraštruktúre. Následne sme vyhodnotili jednotlivé služby z pohľadu ich efektívnosti, zložitosti zakomponovania do cloudu, prípadne iných kritérií. Metóda skúmania bola zameraná teda na experimentálnom odskúšaní prvkov pre poskytovanie služby kybernetickej bezpečnosti v niekoľkých pripravených scenárov použitia v izolovanom cloudovom prostredí.

Posledná fáza pri našej metodike SDLC je údržba. Tú budú vykonávať administrátori cloudu, ktorí zodpovedajú za správny chod cloudu a služieb na ňom spustených. Venovať sa jej môžu aj študenti rozšírením témy v rámci projektovej výučby alebo pri vypracovávaní súvisiacej záverečnej práce.

3.1 Príprava testovacieho prostredia

Katedra informačných sietí poskytla pre potreby testovania nasadenia SECaaS na cloude OpenStack testovacie prostredie vo forme serverového zariadenia. Server je virtualizovaný a je spustený pod systémom pre správu hardvéru pre virtualizáciu VMware ESXi. Server disponuje celkovo 32GB operačnej pamäte, 16 procesorovými jadrami a 800 GB úložiskom pre dáta. Tieto systémové prostriedky spĺňajú minimálne odporúčané systémové požiadavky pre spustenie základného OpenStack cloudu pomocou nástroja DevStack. Testovacie prostredie je spustené na Ubuntu 20.04, vzhľadom na to, že na tejto distribúcii Linuxu je DevStack najviac odladený [21]. Finálny OpenStack bude vo verzii Victoria, čo je v čase vypracovania práce jeho najaktuálnejšia stabilná verzia.

Zabezpečenie testovacieho prostredia

Naše testovacie prostredie je potrebné zabezpečiť voči kybernetickým útokom, vzhľadom na to, že prístup k serveru, na ktorom beží je otvorený z celého internetu.

V linuxovom jadre je sprostredkovaný systém pre paketové filtrovanie Netfilter. Tradičný spôsob ako manipulovať správanie tohto systému je pomocou Iptables. Iptables je nástroj umožňujúci pridávať filtračné pravidlá do Linuxového kernelového firewallu pre povolenie a blokovanie špecifickej IP prevádzky na koncových zariadeniach. Koncové firewall riešenie implementované využitím Iptables je zároveň vysoko konfigurovateľné a flexibilné.

Prednastavený konfiguračný nástroj pre firewall na Ubuntu systémoch je UFW (Uncomplicated Firewall). UFW [30] bol vyvinutý za účelov zjednodušiť konfiguráciu Iptables na koncových zariadeniach tým, že poskytne používateľovi zjednodušený prístup k správe firewallových pravidiel cez príkazový riadok. Prvý krát bol vydaný s distribúciou Ubuntu 8.04 LTS a odvtedy je súčasťou každého nasledujúceho vydania operačného systému.

Prvým krokom pri zabezpečení servera je teda zapnutie UFW. Pridávané pravidlá sa aplikujú do firewallu ihneď. Pre účely zabezpečenia servera bude možné pristupovať na server len z IP rozsahu prideleného Žilinskej univerzite v Žiline. Taktiež sa obmedzí aj prístup ku cloudu len pre klientov patriacich do tohto rozsahu. Pre vzdialený prístup na server a prístup ku testovaciemu OpenStacku je nutné buď priame pripojenie do univerzitnej siete alebo virtuálne pripojenie pomocou VPN klienta.

Inštalácia cloudu pomcou DevStack

Inštaláciu testovacieho cloudu OpenStack bola realizovaná pomocou nástroja DevStack. Najnovšiu verziu nástroja je možné získať naklonovaním si oficiálneho repozitára dostupného na platforme pre kolaboráciu a spravovanie verzií zdrojových kódov GitHub. Pred spustením samotného skriptu pre inštalovanie cloudu je odporúčané aktualizovať si repozitáre pre balíčkový systém a následne si aj jednotlivé nainštalované balíčky aktualizovať zodpovedajúcimi shell príkazmi.

Sieťová konfigurácia cloudu

Dôležitá úloha pri nasadzovaní cloudu je správna konfigurácia sieťovania. DevStack ponúka niekoľko možností pre prispôsobenie siete inštalovaného OpenStacku pre zaručenie funkčnosti konektivity inštanciám strojov používateľov v cloude. Pri nešpecifikovaní explicitného nastavenia siete sa vytvorí cloud, ktorý umožní klientom z vnútra sa pripájať do externého sveta mimo cloudu.

Po inštalácií, sa na cieľovom zariadení objaví nové virtuálne rozhranie br-ex, ktoré je spravované Neutron službou. Toto rozhranie je napojené do siete 172.24.4.0/24 s adresou

172.24.4.1. V cloude vystupuje ako verejná sieť smerom do externého sveta, a taktiež sa používa ako rozsah pre prideľovanie plávajúcich adries pre prístup k inštanciám. Tým pádom, je prostredie cloudu izolované a je možné k nemu pristupovať len zo samotného stroja, na ktorom je spustený. V prípade, že si nasadenie vyžaduje konektivitu ku klientom z LAN siete, ku ktorej je server pripojený, je potrebné napojiť br-ex na nejaké fyzické rozhranie. Nastaví sa IP maškaráda na rozhraní smerom von z clodu, aby mohli inštancie pristupovať do internetu.

Všetky OpenStack prostredia majú po dokončení inštalácie nastavené firewallové pravidlá pre zablokovanie všetkých paketov vchádzajúcich do cloudu, ktoré nie sú odpovede na žiadosti inicializované od cloudových inštancií. Z toho dôvodu sa musia pridať do prednastavenej bezpečnostnej skupiny s názvom default pravidlá pre povolenie ICMP a SSH. V prípade, že by bolo v cloude viacero skupín s názvom default (napr. keď je v cloude viac projektov), treba pridať tieto pravidlá do všetkých týchto skupín.

Testovacia topológia

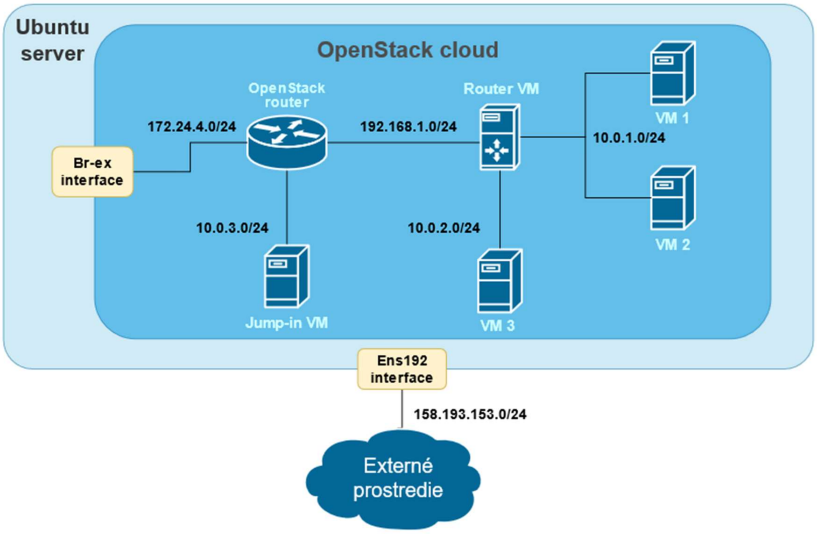
Implementácia a otestovanie služieb zabezpečenia bude prebiehať na topológii pozostávajúcej zo šiestich zariadení, vytvorených na základe HOT skriptu. Pri vstupe do siete sa nachádza OpenStack smerovač. Ďalej v nej vystupuje päť virtuálnych linuxových zariadení, jeden z nich plní úlohu smerovania medzi sieťami ostatných zariadení. Virtuálny stroj Jump-In VM použijeme na vzdialený prístup k ostatným inštanciám v infraštruktúre.

Okrem zadefinovania zariadení bolo potrebné dodať do skriptu vytvorenie statických smerovacích záznamov na OpenStack smerovači pre posielanie IP prevádzky do sietí 10.0.1.0/24 a 10.0.2.0/24 cez priamo pripojeného suseda Router VM.

Keďže má Router VM za úlohu smerovať IP prevádzku medzi jednotlivými sieťami zapneme na ňom funkcionalitu smerovania (IP forwarding). Pakety prichádzajúce na Router VM s cieľovou adresou odlišnou priamo pripojeným sieťam sa budú posielať smerom na OpenStack router. Dodáme teda zodpovedajúci smerovací záznam na danom stroji pre nastavenie prednastavenej brány OpenStack Router.

OpenStack má v sebe zabudovanú prednastavenú anti-spoofing ochranu, ktorá je automaticky nastavená na každom rozhraní inštancií. Táto ochrana spôsobuje, že nemôže prísť na rozhranie paket s cieľovou IP adresou mimo rozsahu podsiete, do ktorej je rozhranie pripojené, a taktiež musí byť cieľová MAC adresa zhodná s adresou rozhrania. Pre umožnenie NFV cez zariadenie Router VM je potrebné vypnúť port security, tým pádom

pakety s cieľovou adresou z iných rozsahov nebudú zahodené Neutron službou. Problém by sa dal obísť aj zadefinovaním dodatočných povolených párov IP a MAC adries (allowed address pairs) nad každým rozhraním zariadenia.



Obrázok 25 – Testovaná virtuálna infraštruktúra v cloude

V tomto stave vypracovania máme vytvorený testovací cloud spolu s infraštruktúrou zariadení pre odskúšanie implementácie SECaaS služieb v OpenStack. Musíme však pred samotným implementovaním SECaaS služieb zaviesť spôsob vzdialeného pristupovania ku inštanciám cloudu. Bez vhodného mechanizmu prístupu ku virtuálnym zariadeniam nebude možné služby nasadiť, odskúšať ani vyhodnotiť.

Vzdialený prístup k zariadeniam

Pri spúšťaní inštancií v cloude je potrebné zadefinovať verejný SSH kľúč pre vzdialenú správu zariadenia. Páry kľúčov môžu byť vytvorené v OpenStacku, prípadne sa dajú importovať už vytvorené páry. V cloude sme si vygenerovali testovací pár kľúčov s názvom „default_key_pair“, ktorého verejný kľúč sa vloží do všetkých virtuálnych strojov v topológii, a pomocou zodpovedajúceho privátneho kľúča bude umožnené vzdialené prihlasovanie.

Cloudové zariadenie pripojené na verejné siete teda bude plniť úlohu vstupu do siete, preto ho označíme ako Jump-in VM. V prípade, že by bol cloud otvorený do sveta a mal by

pridelenú verejnú adresu, dal by sa preskočiť krok prihlasovania na server cloudu. Táto adresa by sa mala priradiť zariadeniu určeného pre sprístupnenie zariadení v cloude, v našom prípade Jump-in VM. Ďalej je možné zjednodušiť spôsob prihlasovania na inštancie spustené v cloude implementovaním presmerovania L4 portov na Jump-in VM. Nakonfigurovalo by sa pridelenie jedinečného portu každému zariadeniu a vzdialene by sa pristupovalo kombináciou portu a verejnej adresy cloudu.

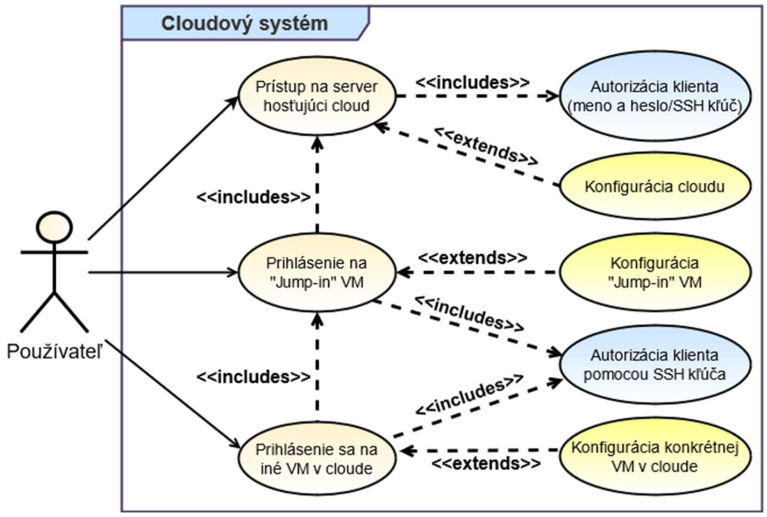
Toto pripojenie Jump-in VM je umožnené v OpenStacku realizovať dvomi spôsobmi. Prvý je explicitné pripojenie nejakého pripojeného zariadenia do verejnej siete medzi cloudom a serverom. IP adresa tohto rozhrania môže byť fixne zadefinovaná, prípadne sa pridelí z rozsahu dostupných adries. Druhý spôsob, ktorý využijeme pri našej topológii, je vytvorenie pseudo-prepojenia zariadenia do verejnej siete cez plávajúce adresy.

Každá spustená inštancia v cloude má priradené virtuálne rozhrania, cez ktoré komunikujú s ostatnými zariadeniami. Tieto rozhrania majú fixnú IP adresu z určitého privátneho rozsahu adries a priradené sú až do vymazania zariadenia. Smerovače v OpenStacku umožňujú vytvoriť statické mapovanie privátnej adresy na verejnú plávajúcu IP adresu, a tým sprístupniť zariadenia aj do mimo-cloudovým klientom. Podmienkou priradenia plávajúcej IP adresy rozhraniu inštancie je, že dané rozhranie musí byť priamo pripojené k OpenStack smerovaču. Plávajúce IP adresy sú spravované v cloude Nova výpočtovou službou. V jednom čase je možné aby mala inštancia priradenú len jednu plávajúcu IP adresu. Prednastavene vytvorí DevStack v OpenStacku rozsah plávajúcich adries od 172.24.1.2-172.24.1.254.

Hlavnou výhodou plávajúcich adries voči fixným je to, že môžu byť flexibilne odobraté a priradené medzi inštanciami v cloude, zatiaľ čo fixná adresa je priradené do konca životného cyklu inštancie. Design v cloudoch by sa mal riadiť modelom, že všetko by malo byť čo najviac disponibilné. Dodatočné rozhranie určené len pre externý prístup taktiež pridáva zbytočnú dodatočnú virtualizačnú záťaž na cloud.

V testovacej topológii pridelíme plávajúcu adresu rozhraniu na Jump-in VM priamo pripojeného do siete smerom na smerovač. Zadefinujeme si, aby pridelilo pri každom vytvorení topológie vždy IP adresu 172.24.4.10. Po prihlásení sa na zariadenie Jump-in VM

pripojenej do verejnej siete cez plávajúcu IP adresu je možné vzdialené prihlasovanie na všetky ostatné zariadenia v infraštruktúre.



Obrázok 26 – Diagram prípadov použitia pri pristupovaní ku cloudu

Pri prihlasovaní sa z Jump-in VM na ostatné inštancie v cloude je nakonfigurované, že je potrebný rovnaký SSH kľúč ako v prípade prvotnom vzdialenom prihlásení. To znamená, že ak chceme pristupovať k ostatným inštanciám, musíme vložiť privátny kľuč aj na Jump-in VM. Tento postup nie je odporúčaný, pretože privátne kľúče by mali ostať len u koncového používateľa a nemali by byť zdieľané.

Riešenie bezpečnostnej hrozby je preposlanie privátneho kľúča z SSH klienta na vzdialené zariadenie, ktoré môžeme docieliť použitím SSH agent-forwarding mechanizmu [31]. SSH agent je pomocný program pre udržiavanie si informácií o používateľských kľúčov a prihlasovacích údajov v operačnej pamäti zariadenia. Umožňuje aj vzdialeným serverom autentifikovať používateľa jeho lokálnymi kľúčmi aj v prípade, že je vzdialené prihlásený na iné zariadenia. Privátne kľúče používateľa sa teda neposielajú na prihlásené zariadenia, ani nie je potreba ich šifrovať. Mechanizmus len sprostredkuje spôsob vzdialenému serveru, na ktorý sa prihlasuje, autorizovať klienta jeho SSH kľúčom uloženého lokálne.

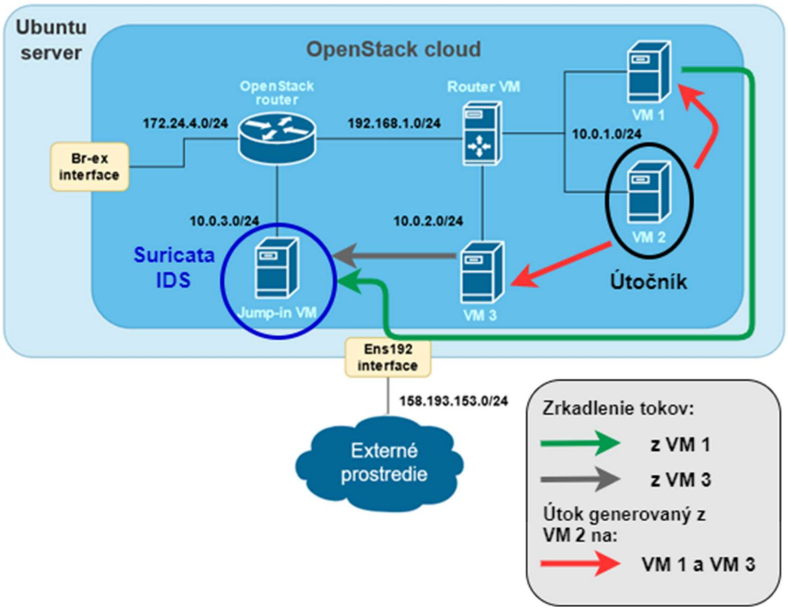
****** 4VYSLEDKY**

4.1 Implementácia cloudovej IDS/IPS služby

Otestované boli dva scenáre. Prvý scenár bol určený pre spustenie IDS na Jump-in VM, na ktorý sa budú zrkadliť prijímané IP toky klientov SECaaS služby v cloude. Zrkadlenie prevádzky bude zabezpečovať služba TAPaaS v OpenStacku. Pri druhom scenári sa nasadí na Router VM transparentný IPS, cez ktorý bude prechádzať prevádzka cloudových inštancií. Pri identifikácií škodlivej prevádzky IPS zablokuje zodpovedajúce pakety a ochráni tak klientov SECaaS služby pred útokmi škodlivých strán.

4.1.1 Scenár 1 – Sieťový IDS cez TAPaaS

Tento scenár je zameraný na implementovanie sieťového IDS systému využitím Neutron služby TAPaaS pre zrkadlenie prevádzky rozhraní v cloude. IDS bude spustené na virtuálnej inštancií Jump-in VM. Zadefinujeme si, že používateľmi SECaaS služby budú inštancie VM 1 a VM 3. Je potrebné nakonfigurovať, aby sa v cloude zrkadlila prevádzka prijímaná na sieťových rozhraniach týchto zariadení. Napokon sa vyhodnotí funkčnosť služby vykonaním útokov na VM1 a VM3. Útoky budú generované z VM 2. V tomto scenári nechránime aktívne klientov služby, len sa detekuje vznik bezpečnostných incidentov.



Obrázok 30 – Implementácia IDS služby v infraštruktúre

Konfigurácia TAPaaS

Existujú 2 spôsoby ako pracovať so službou TAPaaS, buď pomocou Neutron CLI klienta alebo cez Horizon webové rozhranie, pri ktorom musí byť nainštalované rozšírenie TAPaaS-dashboard. My budeme používať pri tvorbe služby CLI klienta. Pri konfigurácií TAPaaS služby sa pracuje s TAP službou a TAP flow.

Prvý krok je vytvorenie TAP služby, ktorá bude slúžiť ako cieľ pre zrkadlenú prevádzku. Povinný parameter pri zadávaní príkazu pre tvorbu TAP služby je identifikátor rozhrania, na ktoré sa majú zrkadliť pakety. Následne sa vytvárajú TAP flow objekty označujúce rozhrania, ktoré chceme zrkadliť. Je potrebné ich pridružiť už vytvorenej TAP službe. Povinné parametre sú teda rozhranie, názov TAP služby a smer prevádzky, ktorý nadobúda hodnoty buď IN (pakety prijímané na rozhraní), OUT (pakety poslané z rozhrania), alebo BOTH (všetka IP prevádzka). Ako bolo uvedené, v riešení budú SECaaS službu odoberať inštancie VM 1 a VM 3. Pred potencionálnymi hrozbami chceme ochrániť konečné stroje a nie je dôležité čo dané inštancie posielajú von, preto budeme zrkadliť len IP prevádzku prijímanú na ich rozhraniach.

Overenie zrkadlenia do cieľovej inštancie môžeme overiť vykonaním jednoduchého testu s príkazom ping.

Konfigurácia Suricata IDS

Inštalácia Suricaty a správa potrebných systémových knižníc a balíčkov bola vykonaná na základe postupov popísaných v práci [32], a podľa oficiálnej dokumentácie [33]. Proces inštalácie sme spustili pomocou programu APT pre správu systémových balíčkov príkazom „apt install suricata“, je potrebné si však do systému pridať dodatočný balíčkový archív udržiavaný nadáciou pre otvorenú bezpečnosť informácií Open Information Security Foundation (OISF) príkazom „add-apt-repository ppa:oisf/suricata-stable“. Správu služby vykonáva v Linuxe Systemd manažér systému, ktorý umožňuje modifikovať chod služby pomocou API volaní do operačného systému. Konfiguráciu spúšťania a behu služby sa uskutočňuje v súbore „/etc/default/suricata“. Tam je potrebné zadať názov sieťového rozhrania, na ktorom bude prijímaná prevádzka používateľov IDS služby.

Nakonfigurovanie Suricaty sa vykonáva v súboroch uložených v YAML formáte. Je potrebné v konfigurácii zadefinovať domácu sieť, pre ktorú má IDS identifikovať potencionálne hrozby, a potom externú sieť, z ktorej očakávame kybernetické útoky. V súboroch obsahujúcich signatúry pre Suricatu zapneme (respektíve vypneme) jednotlivé pravidlá pre detekovanie hrozieb a dodatočne môžeme zadefinovať určité výstupy z programu vo forme štatistík, logov, a iné [32]. Detekčné pravidlá pre IP prevádzku sa definujú v inom súbore, na ktorý sa odkazuje v hlavnom konfiguračnom súbore.

Pri prvom scenári tvoria domácu sieť koncový používatelia SECaaS služby (VM 1, VM 3) a za externú sieť považujeme všetky ostatné. Definovanie používateľských IDS pravidiel budeme vykonávať vo vlastnom súbore „/var/lib/suricata/rules/my.rules“, ktorý je umiestnený v prednastavenom priečinku pravidiel pre Suricatu. Na tento súbor je potrebné sa odkazovať v konfigurácii IDS. Pre overenie funkčnosti systému si tam zadefinujeme dve pravidlá. Prvé bude kontrolovať pokusy o vzdialený prístup cez protokol SSH z externej siete. Druhé pravidlo bude slúžiť pre detekovanie pokusu o Denial of Service (DoS) útok na port HTTP do internej siete.

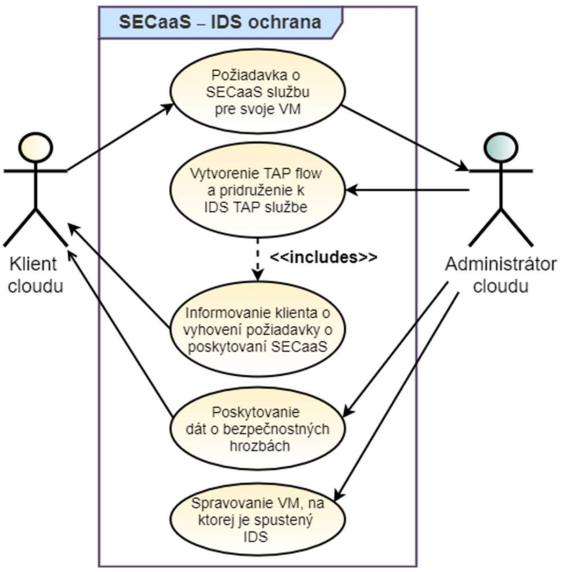
Otestovanie funkčnosti IDS a vyhodnotenie

Aplikujeme konfiguračné zmeny reštartom služby. Následne sa pokúsime o vzdialený prístup na klientov služby zo zariadenia mimo domácej siete, napr. VM 2. Suricata v prípade, že funguje a je nakonfigurovaná, zachytí tento pokus a vygeneruje zodpovedajúci logovací záznam. Otestovanie monitorovania potencionálneho pokusu o odopretie webovej služby bežiacej na VM 1 v domácej sieti je možné cez penetračný nástroj hping3 [34] pre generovanie škodlivej IP prevádzky. Za zdroj generovaných paketov vyberieme opäť VM 2.

Úspešne sa odhalil pokus o vzdialený prístup mimo zadefinovanej siete a taktiež vo výpise môžeme vidieť DoS SYN útok na HTTP službu smerovaný na VM 1. Prvé dva výpisy v logu sa týkajú prevádzky určenej pre zariadenie, kde je spustené IDS. Vzhľadom na to, že Jump-in VM nie je zaradené do domácej siete a použili sme ho na vzdialené prihlásenie na VM 2 (zdroj generovanej prevádzky), IDS vyhodnotil tieto pakety za potencionálnu hrozbu. Záznam v logu obsahuje okrem popisu identifikovanej hrozby aj zdroj/cieľ, sieťové porty, klasifikačný popis, prioritu a časový údaj o momente detekcie ohrozenia. V tomto kontexte predstavuje nižšia priorita väčšiu závažnosť hrozby.

Výstupy generované IDS by sa dali ďalej preposielať na centrálne úložisko pre logy, ktoré používajú iné programy pre rôzne analýzy. Suricata v režime IDS nevykonáva zablokovanie škodlivej prevádzky. Používanie IDS je napríklad pre sofistickejšiu analýzu útokov, ktoré dokážu obísť tradičnú firewallovú ochranu.

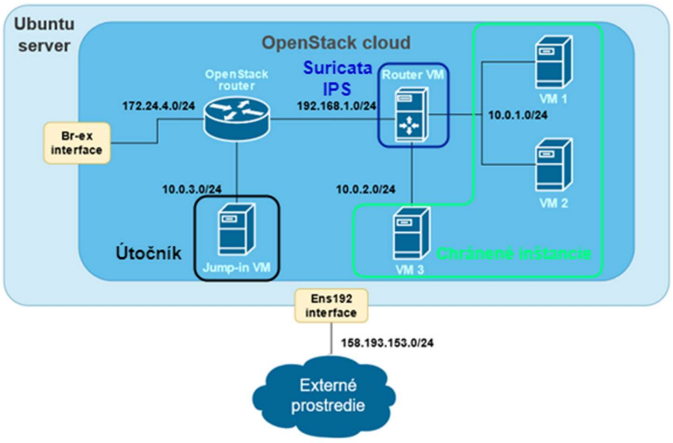
Takáto podoba poskytovania SECaaS formou sieťového IDS vie byť v produkcii zabezpečená nasadením dedikovaného virtuálneho zariadenia v cloude, na ktorom bude spustená Suricata. Domácu sieť budú tvoriť privátne IP používanie v cloude, externá sieť bude všetko ostatné. Požiadavky klientov cloudu o odber SECaaS služby budú smerované na administrátora cloudu, ktorý vytvorí TAP flow pre ich konkrétne zariadenie a pridruží ho IDS TAP službe.



Obrázok 39 – Diagram prípadov použitia cloudového IDS

4.1.2 Scenár 2 – Transparentný IPS

Druhý scenár otestovania nasadenia SECaaS služby v izolovanom cloude je zameraný na spustenie IPS pre zavedenie aktívnej ochrany voči prichádzajúcim útokom. IPS sa bude spúšťať na dedikovanej VM v cloude. Táto VM musí byť strategicky umiestnená tak, aby monitorovala prevádzku ostatných inštancií. Ostatné VM sa pripájajú priamo na IPS inštanciu. IPS VM je pripojené do viacerých sietí a preto musí mať zapnutú funkcionalitu IP smerovania. V našej virtuálnej topológií bude Suricata v IPS režime umiestnená na inštancii Router VM. Táto VM bude chrániť ostatné inštancie, ktoré sú ku nej priamo pripojené (VM 1,VM 2, VM 3). Overenie funkčnosti IPS systému je nutné vykonávať generovaním útokov z prostredia mimo chránených inštancií, preto útoky budú inicializované z Jump-in VM.



Obrázok 40 – Implementácia IPS služby v infraštruktúre

Výhodou Suricaty je, že sa dá prepnúť do režimu IPS. V tomto režime sa aktívne blokuje IP prevádzka vyhodnotená za potencionálne škodlivú. Oproti základnému režimu vnáša IPS mód určité oneskorenie, vzhľadom na to, že je nutné každý prijatý paket skontrolovať a vyhodnotiť, či sa pustí ďalej alebo sa zablokuje. V tomto režime spolupracuje Surica s Iptables cez Net Filter Queue (NFQ). NFQ je modul Iptables, ktorý umožňuje prenechať rozhodnutie o zahodení (resp. povolení) paketu bežiacemu softvéru.

Pri prvom scenári sme nakonfigurovali signatúry manuálne, ale pri tomto scenári sme to vykonali cez nástroj „suricata-update“. IPS používal sadu pravidiel Emerging Threats (ET) open ruleset. Aktivované boli všetky kategórie prebraných pravidiel.

Konfigurácia Suricata IPS

Inštalácia Suricaty pre implementáciu IPS systému sa podstatne nelíši od klasickej inštalácie popísanej v predošlom scenári. Vyžaduje sa, aby mala Suricata zakomponovanú podporu pre NFQ, čo sa dá overiť zadaním „suricata –build-info | grep NFQueue“.

Suricata v režime IPS kontroluje prevádzku z NFQUEUE. Aby sme potrebnú prevádzku presmerovali do NFQUEUE, budeme pridávať pravidlá do iptables.

Po presmerovaní prevádzky do NFQ prebehne IPS kontrola Suricatou. Keďže inštancia, na ktorej spúšťame IPS v infraštruktúre, plní úlohu smerovača, zameriame sa na monitorovanie IP prevádzky, ktorej cieľom sú iné inštancie. Presmerovanie cez Iptables do

NFQ zabezpečíme pridaním pravidla „iptables -I FORWARD -j NFQUEUE“. Ak by mal IPS chrániť aj samotnú inštanciu, na ktorej je spustené, treba ešte zadať „iptables -I INPUT -j NFQUEUE“. Následne stiahneme signatúry ET open pre IPS cez „suricata-update“, čo aktivuje samotný systém.

Otestovanie funkčnosti IPS a vyhodnotenie

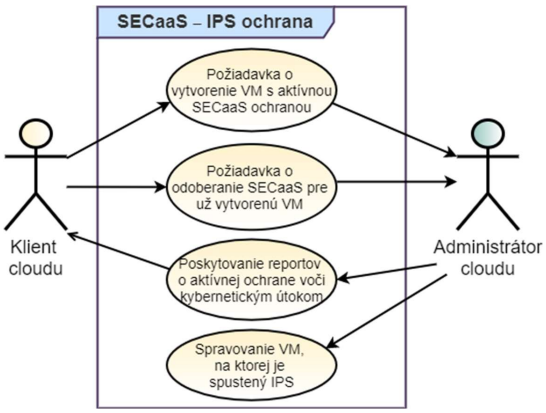
Pri otestovaní sa bude používať rovnaký DoS útok ako pri prvom scenári, teda generovanie veľkého počtu SYN paketov na port 80 s náhodnou zdrojovou IP adresou. Tento útok bude trvať 5 minút a vykonáme ho dva krát, prvý krát bez spustenia IPS a potom zapneme IPS. Budeme sledovať ako sa bude vyvíjať zaťaženie systému v čase. Na VM 1 sme si doinštalovali apache2 server, aby mohol odpovedať na SYN pakety.

Keďže sme spúšťali útok na 5 minút zameriame sa na druhú hodnotu. Pri prvom útoku bez nasadeného IPS bolo priemerné zaťaženie systému 81%. Po zapnutí IPS a opakovaní útoku klesla priemerná záťaž na hodnotu 23%. Zavedenie IPS prvku spôsobilo, že počas útoku mal VM 1 zníženú záťaž o 58%.

Pri druhom pokuse, kedy bol zapnutý IPS sme sledovali koľko paketov prejde na cieľové zariadenie. Útok celkovo vygeneroval 561 tisíc paketov. Z týchto paketov bolo pravdepodobne z dôvodu zahltenia virtuálnej linky prijatých na zariadení Router VM dokopy 496 tisíc. IPS zdetekoval prebiehajúci DoS útok a aktívne zablokoval približne 491 040, čo je 99% škodlivých IP paketov.

Informácie o detekcii a zahodení škodlivých paketov je možné vidieť aj v logoch Suricaty na Router VM. Stiahnuté signatúry ET open úspešne detekovali prebiehajúci útok a zablokovali ho.

Týmto scenárom sme zistili, že je možné poskytovať SECaaS službu pomocou sieťového IPS systému. IPS je nasadený na strategicky umiestenú VM v cloude, cez ktorú bude prechádzať IP prevádzka ostatných VM. Priradenie služby klientovi sa vykonáva pri vytvorení jeho VM tým, že ich napojíme na inštanciu, na ktorej beží IPS. Poskytovanie IPS služby pre už vytvorenú VM v cloude dokážeme realizovať prekonfigurovaním infraštruktúry tak, aby bola VM priamym susedom IPS VM.



Obrázok 44 – Diagram prípadov použitia cloudového IPS

4.2 Implementácia služby pre posudzovanie zraniteľností cloudu

V práci [28] je popísaný postup použitia nástrojov pre posudzovanie zraniteľností pre implementáciu služby pre vykonanie bezpečnostného auditu v OpenStack cloude. Takýto typ služby sa označuje ako Vulnerability Assesment as a Service (VAaaS), v preklade posudzovanie zraniteľností ako služba.

V tomto experimente sme sa pokúsili o implementáciu VAaaS služby v izolovanom cloude, cez ktorú bude možné skenovať cloud, napr. pre zistenie zraniteľností bežiacich inštancií, spustené služby používateľov IaaS/PaaS, a podobne.

Spomedzi rôznych skenovacích nástrojov sme si pre implementáciu služby vybrali nástroj OpenVAS, pretože podľa [28] produkoval podrobné a dobre štruktúrované výstupy.

Za vývojom OpenVAS stojí nemecká spoločnosť Greenbone networks. Firma uskutočnila v roku 2019 prechod, pri ktorom zmenila názvy svojich softvérových riešení, aby obsahovali aj názov spoločnosti pre zvýšenie pozornosti verejnosti. Preto sa po novom funkcionalita OpenVAS zakomponovala do iného nástroja s názvom GVM (Greenbone Vulnerability Management) prechodom z OpenVAS 9 na GVM 10.

Greenbone Vulnerability Management [36]

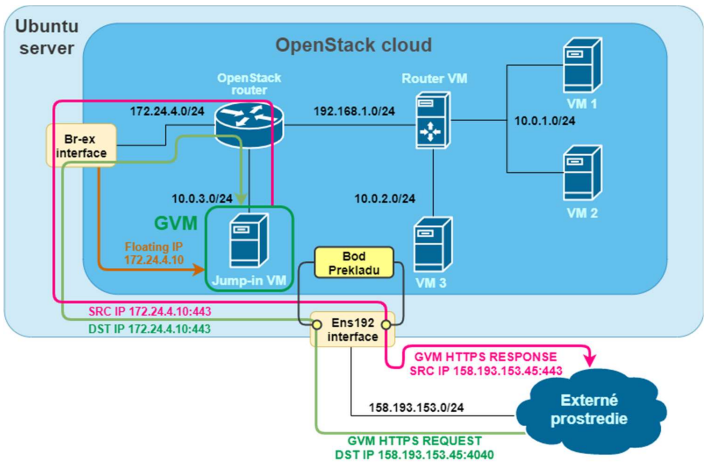
GVM je open source nástroj od Greenbone Networks, zameraný na lokalizáciu akýchkoľvek zraniteľnosti existujúcich v IT infraštruktúre, vyhodnotenie ich závažnosti a poskytnutie odporúčaní pre odstránenie (príp. zmiernenie) bezpečnostného rizika.

Výhodou nástrojov pre správu zraniteľností je automatizácia, nastavenie časového plánu skenovania a možnosť využitia prispôsobených konfigurácií pre skenovanie infraštruktúry. Nástroje od spoločnosti Greenbone sú vyvíjané v súlade so štandardmi pre dosiahnutie najvyššej úrovne dátového zabezpečenia (napr. General Data Protection Regulation – GDPR) [36].

4.2.1 Úprava zabezpečovacieho skriptu

Služba pre skenovanie cloudovej infraštruktúry bola spustená na zariadení Jump-in VM. Momentálne nastavenie firewallu na serveri, na ktorom beží OpenStack CC, nepovolí prístup ku službe. Musíme modifikovať skript pre zabezpečenie cloudového prostredia tak, aby prevádzka s určitým portom smerovaná na verejnú adresu cloudového servera bola preložená na plávajúcu IP adresu pridelenú Jump-in VM a HTTPS port 443.

Cloudové prostredie sme zabezpečili za pomoci nástroja UFW, ktorý sme opisovali v časti Zabezpečenie testovacieho prostredia. Tento nástroj zjednodušuje pridávanie pravidiel na povolenie (resp. blokovanie) prevádzky na základe sieťového portu alebo konkrétnych rozsahov IP adries. Pre implementovanie komplexnejšieho správania firewallu (napríklad presmerovanie portov, preklady IP adries) však musíme upraviť súbor „/etc/ufw/before.rules“ s pravidlami napísanými v Iptables syntaxi. Najskôr sa pri reštartovaní firewallu načítajú tieto pravidlá a následne sú pridané UFW pravidlá. Výsledné správanie je teda také, že prednosť pri vyhodnocovaní majú pravidlá pridané do súboru pred UFW pravidlami. V skripte pre zabezpečenie cloudu sme nastavili aby HTTPS prevádzka na verejnú adresu servera s cieľovým portom 4040 sa presmerovala na Jump-in VM cez pridelenú plávajúcu IP adresu s portom 443. Zabezpečili sme to prekladom cieľovej IP a cieľového portu, viď obrázok Obrázok 45 a **Chyba! Nenašiel sa žiaden zdroj odkazov.**.



Obrázok 45 – Sprístupnenie GVM v infraštruktúre do internetu

4.3. Zhrnutie

Výskumom sme analyzovali možné nasadenie SECaaS v CC OpenStack, popísali sme zabezpečenie testovacieho prostredia určeného pre nasadenie a odskúšanie vybraných SECaaS služieb. V testovacom cloude sme vytvorili niekoľko rôznych SECaaS služieb. Služby sme otestovali a vyhodnotili výhodnosť ich používania v diskusií k výsledkom.

4.3 Zisťovanie zraniteľností v OpenStacku využitím interných a externých nástrojov

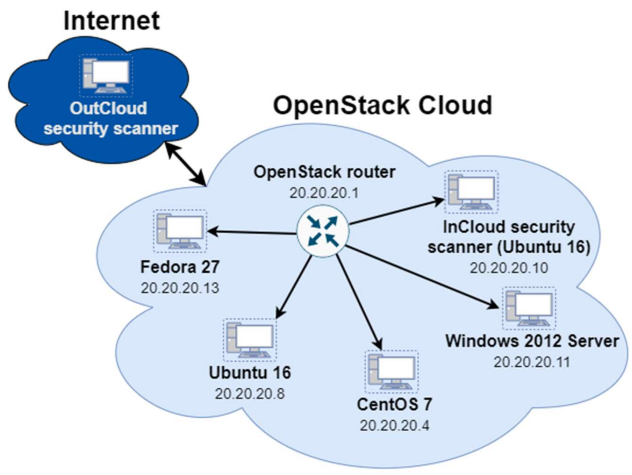
Autori v článku [28] uskutočnili analýzu bezpečnosti pri používaní privátneho OpenStack cloudu z vnútorného a vonkajšieho prostredia pomocou softvérových nástrojov určených na skenovanie sietí pre odhalenie zraniteľností: Nessus, Metasploit, OpenVAS.

Cloudová infraštruktúra použitá pre účely odhalenia zraniteľností pozostávala z piatich virtuálnych strojov s odlišnými operačnými systémami, viď obrázok Obrázok 20. Práca opisuje bezpečnostné riziká nájdené v prostredí pre viacerých používateľov a navrhuje riešenia pre ošetrenie týchto rizík. Cieľom posudzovania je analýza rizík OpenStack uzlov a virtuálnych strojov z interného, ale aj z externého prostredia systému.

Nessus je proprietárny skener zraniteľností vytvorený spoločnosťou Tenable Network Security. Je možné využívať aplikáciu bezplatne na testovanie skenovania na maximálne 16 rôznych IP adresách [29]. Nessus ako jeden z mála ponúka skenovací modul špecificky navrhnutý pre OpenStack. Metasploit je skener, ktorý taktiež dáva možnosť vykonať prienik do skenovaného systému využitím identifikovaných zraniteľností. V ponuke je základná bezplatná verzia a platená verzia. OpenVAS je bezplatné riešenie pre skenovanie a odhalenie bezpečnostných rizík.

Vnútro-cloudové skenovanie sa vykonalo pomocou zariadenia umiestneného vnútri OpenStacku, na ktorom je nainštalovaný spomenutý softvér. Skenovanie sa realizovalo aj z externého prostredia zariadením umiestneným mimo internej OpenStack siete cez rovnaké softvérové nástroje.

V práci je uvedené, že Nessus odhalil v externom prostredí 2 závažné bezpečnostné rizika, 3 stredne závažné a 1 bezpečnostné riziko malej závažnosti. Skenovaním z externého prostredia cez Metasploit sa nenašli žiadne bezpečnostné riziká. Ohlásil len zoznam otvorených TCP/UDP portov. Skenovanie pomocou OpenVAS odhalilo rovnaké riziká ako Nessus, no identifikovalo okrem toho ešte jedno málo závažné riziko týkajúce sa TCP časových pečiatok, cez ktoré by bolo možné vypočítať čas prevádzky servera.



Obrázok 20 – Infraštruktúra pre testovanie zraniteľností cloudu [28]

Vnútorným skenovaním sa získal kompletný report o bezpečnosti každého koncového zariadenia nachádzajúceho sa v cloude. Najviac bezpečnostných rizík bolo na

Windows serveri, ktorý mal 11 rizík. Čo sa týka počtu skenovaných portov, tak najviac ich odhalil Metasploit, konkrétne 16. OpenVAS našiel 8 portov a Nessus najmenej s počtom 7.

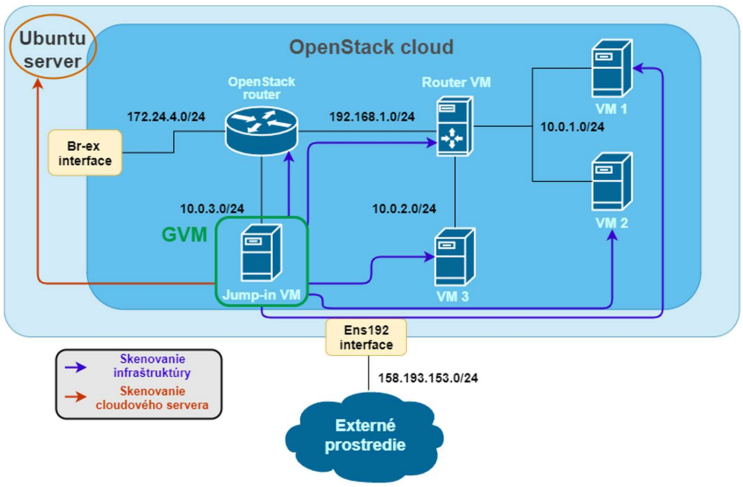
Zhodnotením výsledkov sa zistilo, že Metasploit poskytol najviac výstupov. Nessus zasa poskytol najviac informácii o odhalených portoch a poskytol možné riešenia pre zmiernenie rizika. OpenVAS ako jediné bezplatné riešenie poskytlo výsledky, ktoré boli zároveň dobre organizované a podrobné.

Testovanie GVM as a Service (GVMaaS) a vyhodnotenie

Testovanie poskytovania SECaaS vo forme cloudovej služby pre posudzovanie rizík cez nástroj GVM prebiehalo vykonaním skenovania interných VM spustených vo vytvorenej infraštruktúre. V prípade, že je nasadená IPS služba, je potrebné ju buď vypnúť alebo zaradiť Jump-in VM do domácej siete v konfigurácii Suricaty, aby IPS nezablokoval prevádzku generovanú nástrojom GVM.

Operačný systém VM v infraštruktúre je Ubuntu 20.04. Doinštalovali sme niekoľko služieb na zariadenia a sledovali sme, či skener odhalí aké služby sú na jednotlivých VM spustené. Na VM 1 sme nainštalovali Apache2. Na Router VM sme spustili XRDP (X Remote Desktop Protocol) službu spolu s desktopovým GNOME (GNU Network Object Model Environment) prostredím. Na záver sme nasadili na VM 2 open source nástroj FreeRADIUS pre centralizovanú správu autorizácie, autentifikácie a účtovania. Na VM 3 neboli doinštalované žiadne dodatočné balíky.

Spustili sme cez webové rozhranie skenovanie všetkých zariadení v infraštruktúre a cloudového servera, viď obrázok Obrázok 48. Pred spustením sa volí parameter kvality skenovania (Quality of Detection – QoD), ktorý vyjadruje ako spoľahlivý je výsledok skenovania a jeho hodnota sa udáva v percentách. Čím vyšší QoD, tým menšia pravdepodobnosť, že bude identifikovaná hrozba predstavovať falošne pozitívne riziko. Zvýšením hodnoty QoS teda dosiahneme presnejšie výsledky [38]. Pre skenovanie použijeme prednastavenú hodnotu 70%. Zadefinovali sme, aby sa vykonalo aj skenovanie samotného servera, na ktorom je spustený OpenStack.



Obrázok 48 – Skenované zariadenia nástrojom GVM

Skenovanie sa skončilo približne po 2 a pol hodiny. Čas potrebný pre vykonanie skenovania závisí od počtu skenovaných VM. Úspešné sa identifikovala verzia operačného systému všetkých VM. Najviac hrozieb bolo identifikovaných na zariadení hosťujúcom cloud. Zároveň bolo na ňom identifikovaných najviac otvorených portov. Najzávažnejšie riziko sa našlo na zariadení, kde je spustené GVM (Jump-in VM). Týkalo sa to ponechania prednastaveného mena a hesla pri prihlasovaní sa do webového rozhrania GSA. Taktiež sa identifikovalo, že toto webové rozhranie používa zastaralú verziu TLSv1 pre šifrovanie webovej prevádzky. Výstup z programu nám poskytol informáciu, že protokol VNC (Virtual Network Computing), ktorý OpenStack používa pre poskytovanie vzdialeného grafického prístupu k inštanciám v cloude nepoužíva žiadnu autentifikáciu. Toto nepredstavuje riziko v našej infraštruktúre, pretože pre prístup na inštancie v cloude používame SSH kľúče a nie je možné sa prihlásiť na inštancie bez adekvátneho privátneho kľúča. Na cloudovom servery sa našlo niekoľko stredne závažných rizík tykajúcich sa verzie úložiska Etcd 3.3.12. Tieto riziká by sme dokázali odstrániť prechodom na novšiu verziu daného softvéru. Okrem toho sme zistili, že je možné pristupovať k určitým súborom používaným GIT-om umiestneným na cloudovom serveri. Posledné identifikované riziko nízkeho stupňa sa týka získania TCP časových pečiatok, cez ktoré je možné vypočítať dobu prevádzky zariadení.

Zoznam identifikovaných služieb v infraštruktúre môžeme vidieť v tabuľke Tabuľka 4. Okrem FreeRADIUS spustenom na VM 2 objavil skener všetky bežiace služby. Absencia

UDP portov 1812 a 1813 používaných nástrojom FreeRADIUS je spôsobená tým, že prednastavený zoznam portov používaný pri skenovaní neobsahuje dané porty. Je možné vytvoriť vlastné zoznamy, v ktorých si administrátor môže zadefinovať, ktoré bežiace služby chce, aby boli skenovaním infraštruktúry odhalené.

Výsledky skenovania sme si cez webové rozhranie exportovali do PDF (Portable Document Format) súboru, ktorý sa nachádza v prílohách záverečnej práce.

Tabuľka 4 – Zoznam identifikovaných portov v infraštruktúre

Identifikovaný port	L4 Protokol	Poznámka	Zariadenie
443	TCP	HTTPS	Jump-in VM
3389	TCP	xRDP vzdialená plocha	Router VM
80	TCP	HTTP	VM 1, Jump-in VM, OpenStack cloud server
22	TCP	SSH	Všetky zariadenia
5900	TCP	VNC pre VM v OpenStacku	OpenStack cloud server
5901	TCP	VNC pre VM v OpenStacku	
5902	TCP	VNC pre VM v OpenStacku	
5903	TCP	VNC pre VM v OpenStacku	
5904	TCP	VNC pre VM v OpenStacku	
2380	TCP	Etcd vo verzii 3.3.12	
6080	TCP	VNC proxy pre OpenStack	
4369	TCP	Služba peer discovery pre RabbitMQ	
5672	TCP	AMQP server	
8775	TCP	OpenStack metadata port	
9696	TCP	OpenStack networking	
25672	TCP	RabbitMQ	

5 Diskusia k výsledkom

Vo fáze realizácie návrhu sa nám úspešne podarilo implementovať tri rôzne SECaaS služby a odskúšať ich na vytvorenej topológii v testovacom OpenStack cloude, ktorú ilustruje obrázok Obrázok 25.

Pri vypracovávaní riešenia sme začali študovaním prác týkajúcich sa bezpečnostných služieb implementovaných v CC OpenStack z databázy technickej literatúry IEEExplore. Zistili sme, ako sa dajú SECaaS služby poskytovať v OpenStacku. Ďalej sme zabezpečili server poskytnutý katedrou KIS pre spustenie testovacieho cloudu pomocou firewallu UFW. Pokračovali sme inštalovaním vlastného OpenStack cloudu cez DevStack skripty.

Pred samotným vytvorením SECaaS služieb sme si vybrali aký typ zabezpečenia budú poskytovať. Na základe konzultácie s vedúcou práce sme rozhodli, že pre katedru KIS bude najvýhodnejšie, ak bude SECaaS poskytovaný formou IDS/IPS systému pre detekciu/blokovanie útokov na cloud a ak bude poskytovať možnosť vykonania bezpečnostného auditu VM spustených v cloude, konkrétnejšie zrealizovaním testu zraniteľností. Na základe týchto požiadaviek sme zostavili tri scenáre potencionálnych SECaaS služieb, ktoré sme následne implementovali a odskúšali v privátnom cloude.

Prvý scenár bol zameraný na vytvorenie centralizovaného sieťového IDS pre detekciu útokov na chránené inštancie v infraštruktúre, pričom sa nasadila a využila služba TAPaaS. V tomto scenári sme dokázali útoky detekovať, škodlivá prevádzka nebola zablokovaná a pokračovala ďalej na koncové zariadenia (VM).

Druhý scenár obsahoval nasadenie sieťového IPS, ktorý spomenutú škodlivú prevádzku dokázal nie len identifikovať, ale aj zablokovať. Pri implementácii IDS/IPS služieb sme použili program Suricata.

Tretí scenár obsahoval službu pre skenovanie zraniteľností v privátom OpenStack cloude, využitý bol nástroj GVM nasadený na jednej vybranej VM v danej topológii (Jump-in VM). Nasedenie tohto nástroja bolo celkom komplikované, keďže pozostáva z niekoľkých komponentov a požadované správanie bolo docielené až po niekoľkých opakovaných pokusoch o inštaláciu. PostgreSQL databázu potrebnú pre nástroj je odporúčané nainštalovať vo verzii 11, pretože pri neskorších verziách databázy je GVM 20.08 značne spomalený. Keďže bolo webové rozhranie pre nástroj spustené v cloude,

museli sme v Iptables nastaviť pravidlá pre preklad IP adresy a portu, aby sme dokázali spúšťať skenovanie mimo cloudu.

6 Záver

Hlavným cieľom výskumu bolo hľadanie možnosti poskytovanie cloudovej služby SECaaS na privátnom OpenStack cloude katedry KIS FRI UNIZA. Neexistuje štandardizovaný postup pre vytvorenie SECaaS služby, preto sme si vybrali metodiku SDLC pre rozloženie si problému na nadväzujúcu postupnosť krokov.

V úvode práce pri analýze súčasného stavu sme sa venovali fázam plánovania a analýzy. Vytýčili sme problém, ktorý bolo potrebné riešiť a taktiež sme čerpali poznatky z noriem týkajúcich sa CC potrebných pre vyriešenie daného problému. Pozreli sme sa na súčasné riešenia SECaaS na trhu, čo nám neskôr pomohlo pri výbere a nasadení služby.

Keďže naša implementácia bola v CC OpenStack, venovali sme sa tomuto systému, jeho slabým a silným stránkam, komponentom a architektúre. Pre implementáciu sme si vybrali také bezpečnostné komponenty, aby čo najviac vyhovovali potrebám a požiadavkám katedry informačných sietí, ktorá prevádzkuje CC OpenStack pre dva svoje základné biznis procesy, ktorými sú vzdelávanie a výskum.

Pre praktické experimenty sme nasadili testovací cloud na serveri poskytnutom katedrou, pomocou nástroja DevStack. Vyriešili sme aj dostatočnú úroveň zabezpečenia tohto servera a vytvorili sme testovaciu topológiu obsahujúcu päť virtuálnych zariadení a jeden OpenStack smerovač. Zaviedli sme spôsob pre vzdialené pristupovanie k jednotlivým zariadeniam v cloude cez jednu inštanciu, ktorej sme pridelili plávajúcu IP adresu. Pri samotnom implementovaní SECaaS služby pomocou vybraných komponentov zabezpečenia, sme zrealizovali tri scenáre. Prvé dva boli zamerané na vytvorenie IDS/IPS služby pre detekciu a blokovanie škodlivej IP prevádzky pomocou nástroja Suricata. Podarilo sa nám pri sieťovom IDS úspešne identifikovať hrozby klientov na základe zrkadlených paketov z ich rozhraní. Sieťový IPS systém umiestnený pred klientami SECaaS služby odhalil vykonané útoky a aktívne zablokoval škodlivú prevádzku. Posledný scenár bol zameraný na zavedenie mechanizmu pre vykonávanie skenovania zraniteľností v cloude. Pre tento účel sme si vybrali jednu VM v cloude, na ktorej sme nasadili nástroj pre testovanie zraniteľností GVM. Cez webové rozhranie sme spustili skenovanie infraštruktúry pozostávajúcej z virtuálnych zariadení. Po ukončení bol vygenerovaný report o službách bežiacich na VM v cloude a obsahoval aj potencionálne hrozby ohodnotené úrovňou závažnosti.

Zoznam použitej literatúry

[1] "Bezpečnost jako služba." [Online]. Available: https://www.systemonline.cz/it-security/bezpecnost-jako-sluzba-z.htm. [Accessed: 14-Apr-2021].

[2] "Security as a Service." [Online]. Available: https://www.microsoft.com/en-us/CloudandHosting/SECAAS.aspx. [Accessed: 14-Apr-2021].

[3] "What is Security as a Service (SECaaS)? | Forcepoint." [Online]. Available: https://www.forcepoint.com/cyber-edu/security-as-a-service-secaas. [Accessed: 14-Apr-2021].

[4] S. Hooda, "Cybersecurity vs. Information Security: Is There a Difference?," 2019. [Online]. Available: https://cloudacademy.com/blog/cybersecurity-vs-information-security-is-there-a-difference/. [Accessed: 26-Apr-2021].

[5] ISO/IEC, "ISO/IEC 17788 -Y. 3500 SERIES Y- Information technology - Cloud computing - Overview and vocabulary," *ITU Series Y*, 2014. [Online]. Available: https://www.itu.int/rec/dologin_pub.asp?lang=e&id=T-REC-Y.3500-201408-I!!PDF-E&type=items. [Accessed: 01-Apr-2021].

[6] "What Is Multicloud? | Multicloud Definition | Cloudflare." [Online]. Available: https://www.cloudflare.com/learning/cloud/what-is-multicloud/. [Accessed: 03-May-2021].

[7] ITU-T, "ITU-T Y.3501 Cloud computing – Framework and high-level requirements," *ITU Series Y*, 2016. [Online]. Available: https://www.itu.int/itu-t/recommendations/rec.aspx?rec=12880. [Accessed: 16-Apr-2021].

[8] ITU-T, "ITU-T Y.3502 Cloud computing - Reference architecture," *ITU Series Y*, 2014. [Online]. Available: https://www.itu.int/rec/T-REC-Y.3502/en. [Accessed: 16-Apr-2021].

[9] ITU-T, "ITU-T X.1602 : Security requirements for software as a service application environments," 2016. [Online]. Available: https://www.itu.int/rec/T-REC-X.1602-201603-I/en. [Accessed: 16-Apr-2021].

[10] "Security as a Service Working Group | CSA." [Online]. Available: https://cloudsecurityalliance.org/research/working-groups/security-as-a-service/. [Accessed: 16-Apr-2021].

[11] “Software Development Company - ScienceSoft.” [Online]. Available: https://www.scnsoft.com/. [Accessed: 16-Apr-2021].

[12] “Manage Cloud Security Risks | AWS, VMware, IBM | HyTrust.” [Online]. Available: https://www.hytrust.com/. [Accessed: 16-Apr-2021].

[13] “Home - CipherCloud.” [Online]. Available: https://www.ciphercloud.com/. [Accessed: 16-Apr-2021].

[14] “Public Cloud Security - Fortinet.” [Online]. Available: https://www.fortinet.com/products/public-cloud-security. [Accessed: 16-Apr-2021].

[15] “Open Source Cloud Computing Infrastructure - OpenStack.” [Online]. Available: https://www.openstack.org/. [Accessed: 20-Mar-2021].

[16] “OpenStack Releases: OpenStack Releases.” [Online]. Available: https://releases.openstack.org/. [Accessed: 20-Mar-2021].

[17] “OpenStack Docs: Introduction: A Bit of OpenStack History.” [Online]. Available: https://docs.openstack.org/project-team-guide/introduction.html. [Accessed: 20-Mar-2021].

[18] “Use Cases - Applications of the Technology - OpenStack.” [Online]. Available: https://www.openstack.org/use-cases/. [Accessed: 20-Mar-2021].

[19] “1.3 OpenStack Nodes.” [Online]. Available: https://docs.oracle.com/cd/E64747_01/E64749/html/osusg-openstack-nodes.html. [Accessed: 20-Mar-2021].

[20] “Three-Node Architecture Overview - Installing and Configuring OpenStack in Oracle® Solaris 11.2.” [Online]. Available: https://docs.oracle.com/cd/E36784_01/html/E54155/archover.html. [Accessed: 20-Mar-2021].

[21] “OpenStack Docs: DevStack.” [Online]. Available: https://docs.openstack.org/devstack/latest/. [Accessed: 20-Mar-2021].

[22] “OpenStack Docs: Deploy tap-as-a-service (TaaS) Neutron / Dashboard plugin.” [Online]. Available: https://docs.openstack.org/openstack-helm/latest/install/plugins/deploy-tap-as-a-service-neutron-plugin.html. [Accessed: 16-Dec-2020].

[23] “OpenStack Docs: Port Mirroring API for VF Mirroring.” [Online]. Available: https://specs.openstack.org/openstack/neutron-specs/specs/stein/port-mirroring-sriov-vfs.html. [Accessed: 16-Dec-2020].

[24] D. L. Rhodes, “The Systems Development Life Cycle (SDLC) as a Standard: Beyond the Documentation.”

[25] “Next Generation Security Services in OpenStack - OpenStack Blog for VMware - VMware Blogs.” [Online]. Available: https://blogs.vmware.com/openstack/next-generation-security-services-openstack/. [Accessed: 09-Dec-2020].

[26] “What is VMware NSX? | Network Security Virtualization Platform.” [Online]. Available: https://www.vmware.com/products/nsx.html. [Accessed: 09-Dec-2020].

[27] C. Xu, R. Zhang, M. Xie, and L. Yang, “Network Intrusion Detection System as a Service in OpenStack Cloud,” in *2020 International Conference on Computing, Networking and Communications, ICNC 2020*, 2020.

[28] I. Gordin, A. Graur, A. Potorac, and D. Balan, “Security assessment of OpenStack cloud using outside and inside software tools,” in *2018 14th International Conference on Development and Application Systems, DAS 2018 - Proceedings*, 2018.

[29] “Download Nessus Vulnerability Assessment | Tenable®.” [Online]. Available: https://www.tenable.com/products/nessus. [Accessed: 21-Apr-2021].

[30] “UncomplicatedFirewall - Ubuntu Wiki.” [Online]. Available: https://wiki.ubuntu.com/UncomplicatedFirewall. [Accessed: 20-Mar-2021].

[31] “Ssh-agent single sign-on configuration, agent forwarding, the agent protocol.” [Online]. Available: https://www.ssh.com/academy/ssh/agent. [Accessed: 11-Apr-2021].

[32] B. Kramár and J. Uramová, “Bakalárska práca - Efektívne ladenie IDS nástroja Suricata pre detekciu známych hrozieb a jeho rozširovanie o nové detekčné metódy,” Žilinská univerzita v Žiline, 2019.

[33] “Suricata User Guide — Suricata 6.0.2 documentation.” [Online]. Available: https://suricata.readthedocs.io/en/suricata-6.0.2/. [Accessed: 13-Apr-2021].

[34] “Hping - Active Network Security Tool.” [Online]. Available: http://www.hping.org/. [Accessed: 14-Apr-2021].

[35] “EmergingFAQ < Main < EmergingThreats.” [Online]. Available: https://doc.emergingthreats.net/bin/view/Main/EmergingFAQ. [Accessed: 11-May-2021].

[36] “Vulnerability Management - Greenbone Networks.” [Online]. Available: https://docs.greenbone.net/. [Accessed: 19-Apr-2021].

[37] “How to Install and Use GVM Vulnerability Scanner on Ubuntu 20.04.” [Online]. Available: https://www.howtoforge.com/how-to-install-and-use-gvm-vulnerability-scanner-on-ubuntu-20-04/. [Accessed: 19-Apr-2021].

[38] “21. Glossary — Greenbone Security Manager (GSM) 6 documentation.” [Online]. Available: https://docs.greenbone.net. [Accessed: 27-Apr-2021].