PROJEKT

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

V1 – Výsledok analýzy dostupných riešení

KPB1 – Návrh virt. riešenia

typ – dokumentácia

mesiac odovzdania – M3

Financované
Európskou úniou
NextGenerationEU

PLÁN OBNOVY

ÚRAD VLÁDY
SLOVENSKEJ REPUBLIKY

Obsah

1 ÚVOD

V súčasnom prostredí neustálych kybernetických hrozieb predstavuje **Multitenantné operačné centrum kybernetickej bezpečnosti (SOC) riešené ako otvorená cloudová služba s prvkami strojového učenia** infraštruktúru pre zlepšenie ochrany digitálnych aktív. Implementácia takéhoto centra si vyžaduje vhodný výber adekvátnej technologickej platformy.

Tento dokument spracováva komplexnú analýzu a porovnanie vybraných open-source riešení, ktoré sú vhodné ako virtualizačné a cloudové platformy pre budovanie zamýšľaného riešenia. Hlavným cieľom je nasadenie multitenantného prostredia, ideálneho pre poskytovanie služieb typu SOCaaS (Security Operations Center as a Service). Analýza je založená na špecifikovaných funkčných, technických, prevádzkových a bezpečnostných kritériách, ktoré boli definované v predchádzajúcom výstupe (V2 – Definovanie výberových kritérií).

Zámerom tejto analýzy je identifikovať a vyhodnotiť silné a slabé stránky analyzovaných riešení. Dôraz je kladený na ich vhodnosť pre nasadenie v akademickom prostredí, ktoré typicky kladie špecifické požiadavky na otvorenosť, flexibilnú škálovateľnosť, vysokú úroveň bezpečnosti, jednoduchú správu a cenovú efektivitu ale zohľadňuje aj potreby VS. Táto analýza poskytnúť pevný základ pre informované rozhodnutie pri výbere vhodnej platformy pre realizáciu projektu.

2 VÝBER KANDIDÁTNYCH RIEŠENÍ

Cieľom identifikácie kandidátskych riešení je vybrať platformu, ktorá dokážu poskytovať zamýšľanú služby typu IaaS (Infrastructure as a Service). Pre nasadenie služby INNOSOC sú potrebné vlastnosti ako vysoká dostupnosť, schopnosť izolácie tenantov a komplexné bezpečnostné mechanizmy. Prednosť dávame riešeniam s otvoreným zdrojovým kódom, ktoré sú vhodné pre akademickú a výskumnú sféru.

2.1 PREHĽAD VHODNÝCH TYPOV RIEŠENÍ

Na základe týchto požiadaviek a predchádzajúcej diskusie sme identifikovali tri kľúčové kategórie riešení, z ktorých vyberieme užší okruh kandidátov pre detailnú analýzu.

2.1.1 Open Source Cloud Management Platforms (CMP) – Plnohodnotné IaaS riešenia

Tieto platformy sú navrhnuté špeciálne pre budovanie a správu privátnych cloudov. Ponúkajú kompletnú sadu služieb pre virtualizáciu výpočtových zdrojov, sieťových funkcií a úlož<iska, vrátane API pre automatizáciu a orchestráciu. Zameriavajú sa na multitenantné prostredia a poskytujú vysokú mieru izolácie. Medzi tieto riešenia radíme:

- **OpenStack** – open-source cloudový systém, ktorý poskytuje široké možnosti škálovania, izolácie tenantov, sieťovej virtualizácie a modulárnosti. Je pre nás základným referenčným riešením.
- **OpenNebula** – Alternatíva s flexibilnou architektúrou, nižšou komplexnosťou a vhodná pre menšie a akademické prostredia.
- **Apache CloudStack** – Robustný a jednoducho škálovateľný.
- **VirtEngine** – Menej známa open-source platforma, ktorá kombinuje správu VM, kontajnerov a účtovanie; má však obmedzenú komunitu a slabšiu podporu.

- **Eucalyptus** – open-source alternatíva k Amazon EC2

Z komerčných sem radíme:

- **Microsoft Azure Stack HCI** – Hybridné riešenie umožňujúce rozšírenie on-premise do verejného Azure cloudu. Vhodné najmä pre Microsoft-centric prostredia.

2.1.2 Hypervisor-centric Virtualization Platforms s prvkami Cloud Managementu:

Ide o riešenia, ktoré primárne slúžia na správu virtuálnych strojov na báze hypervizorov. Ponúkajú webové rozhrania, klastrovacie funkcie a API, ktoré umožňujú efektívnu správu a škálovanie virtuálnej infraštruktúry. Hoci nemusia mať plnohodnotnú "cloudovú" abstrakciu na úrovni IaaS ako CMP, sú mimoriadne výkonné a často jednoduchšie na nasadenie a správu. Tieto riešenia primárne poskytujú virtualizáciu VM, ale sú rozšíriteľné o cloudové funkcie.

- **Proxmox VE** – Jednoducho nasaditeľná, open-source platforma s podporou VM aj kontajnerov. Vhodná pre rýchle nasadenia. Má aktívnu komunitu a vhodné nástroje pre HA a klastrovanie.
- **oVirt** – Enterprise virtualizačná platforma od Red Hatu postavená na KVM. Poskytuje nástroje pre správu virtualizácie, ale menej rozvinuté cloudové funkcie a multitenancy.
- **XCP-ng** - je open-source virtualizačná platforma postavená na hypervízore Xen, určená pre nasadenie VM na fyzický hardvér. V kombinácii s **Xen Orchestra** ponúka kompletné webové rozhranie na správu, zálohovanie a monitorovanie infraštruktúry. Táto dvojica tvorí robustné, škálovateľné a bezlicenčné riešenie vhodné pre firmy aj akademické prostredie.

Z komerčných sem radíme:

- **VMware vSphere** – Jeden z najrozšírenejších virtualizačných systémov s komplexnou podporou pre HA, DR, sieťovanie a správu. Vysoké nároky na licencie.
- **Nutanix AHV** – Hyperkonvergované riešenie, ktoré kombinuje výpočty, úložisko a správu. Silná integrácia, ale uzavretý ekosystém.

2.1.3 Container Orchestration Platforms (s potenciálom pre VM):

Tieto platformy sú dominantné v oblasti kontajnerizácie a mikroslužieb. Hoci nie sú primárne IaaS s podporou VM, ich schopnosti orchestrácie, škálovania a izolácie na úrovni kontajnerov sú relevantné pre moderné architektúry. Niektoré z nich ponúkajú možnosti spúšťania virtuálnych strojov v rámci kontajnerového prostredia. Ich zaradenie je dôležité pre zohľadnenie budúcich trendov a architektúr. V tejto kategórii sú mimoriadne rozšírené riešenia Kubernetes. Ich nasadenie je komplexnejšie a vyžaduje inú architektúru.

- **Kubernetes + KubeVirt** – Štandard pre správu kontajnerov s možnosťou rozšírenia o podporu VM pomocou KubeVirt. Môže byť zaujímavý pri kontajnerizovanom SOCaaS, ale vyžaduje úplne inú architektúru ako klasické IaaS prostredie.
- **Docker Swarm** – natívne orchestrátorské riešenie pre Docker kontajnery, ktoré umožňuje jednoducho vytvárať a spravovať klastre. Umožňuje škálovanie služieb, automatické rozdeľovanie záťaže a vysokú dostupnosť. Vďaka jednoduchému CLI a integrácii s Docker ekosystémom je vhodný pre menšie až stredne veľké nasadenia. Je alternatívou k Kubernetes pre používateľov preferujúcich jednoduchosť.
- **Nomad** – je na prostriedky nenáročný a flexibilný orchestrátor od HashiCorp, určený na nasadzovanie a správu kontajnerových aj nekontajnerových úloh (napr. Binárnych súborov, VM, JAR súbory). Vyznačuje sa jednoduchou architektúrou s jedným

binárnym súborom a vysokou škálovateľnosťou. Podporuje integráciu s Consul a Vault pre service discovery a bezpečnosť. Je vhodný pre tímy, ktoré hľadajú alternatívu ku Kubernetes s nižšou komplexnosťou.

- **Rancher** – open-source platforma na správu Kubernetes klastrov, ktorá zjednodušuje ich nasadenie, monitorovanie a správu z jedného centrálneho rozhrania. Podporuje viacero klastrov naraz, vrátane tých bežiacich v cloude, on-premise alebo hybridne. Poskytuje nástroje na správu prístupov, CI/CD, monitoring a bezpečnosť. Je ideálny pre tímy, ktoré chcú centralizovanú správu Kubernetes bez zložitej konfigurácie.

Z komerčných sem radíme:

- **Red Hat OpenShift Virtualization** – Postavené na Kubernetes s doplnením o podporu VM. Silné pre kontajnerizované aplikácie, menej pre tradičnú IaaS.

2.2 PODMIENKY PRE VÝBER KANDIDÁTOV

Pri výbere kandidátov pre porovnanie v ďalšej fáze projektu boli stanovené tieto základné výberové podmienky:

- Platforma musí byť **open-source**, dostupná pod slobodnou licenciou a bez povinnosti viazania sa na konkrétneho komerčného dodávateľa,
- Musí byť **nasaditeľná on-premise**, s možnosťou prevádzky v prostredí akademickej inštitúcie alebo interného datacentra,
- podpora IaaS funkcionality,
- schopnosť vytvárať multitenantné prostredie,
- jednoduchosť alebo realizovateľnosť nasadenia a správy,
- vhodnosť pre použitie v prostredí bezpečnostnej služby typu SOCaaS.

Cieľom je nájsť riešenie, ktoré poskytujú IaaS (Infrastructure as a Service) funkcionalitu, umožňujú multitenantné prostredie a je dostatočne robustné pre bezpečnostné operácie.

2.2.1 Zúženie výberu

Na základe týchto aspektov boli ako **najvhodnejší kandidáti pre detailnú analýzu** vybrané nasledujúce open-source platformy:

- **OpenStack** [1]**: K**omplexná IaaS platforma s modulárnou architektúrou, širokou podporou virtualizácie, sieťovej segmentácie, izolácie tenantov a automatizácie. Ide o najrozšírenejší open-source cloud s aktívnou komunitou. Výhodou je vysoká flexibilita a integrácia so štandardmi. Nevýhodou môže byť komplexnosť nasadenia.
- **Apache CloudStack** [2]– Robustné, ale menej rozšírené riešenie, s živou ale menšou komunitou a nižšou dynamikou vývoja. Pre tento projekt z pohľadu flexibilnosti nasadenia a ponúkaného ekosystému bol nakoniec preferovaný pred OpenNebula. V prípade jednoduchších akademických projektov by sa to otočilo v prospech OpenNebula [1] [2].
- **Proxmox VE** [5]**:** Virtualizačná platforma s dôrazom na jednoduchosť a efektivitu. Je mimoriadne obľúbená v akademických a testovacích prostrediach. Podporuje VM a kontajnery, disponuje webovým rozhraním a podporou pre klastrovanie. Hoci nejde o plnohodnotné IaaS riešenie, jej architektúra umožňuje niektoré cloudové funkcionality

s menším technickým dlhom. Je však limitovaná v oblasti natívnej multitenantnosti a API orchestrace.

- **Kubernetes** [6]**:** Síce nepodporuje IaaS a VM ale do výberu je zaradeným ako v súčasnosti dôležite riešenie pre beh moderných aplikácií. Ak by bola požiadavka na "kontajnerizované SOCaaS", bol by jednoznačným kandidátom, ale pre "cloud či virtualizačné riešenia" v kontexte VM je menej vhodný.

Nevybraný kandidáti a dôvody vylúčenia:

- **OpenNebula** [7]**:** Jednoduchšia a prístupnejšia alternatíva k OpenStacku. Umožňuje flexibilné nasadenie v menších a akademických prostrediach, podporuje rôzne hypervisory a má rozumnú mieru multitenantnosti, ale na nižšej úrovni. Je menej náročná na inštaláciu aj správu, no poskytuje menej pokročilých funkcionalít než OpenStack či CloudStack. Má menšiu komunitu [1] [2]. Pre nasadenie a prevádzkovanie služieb ŠS je však menej vhodná.
- **oVirt** [8]– Zameraný primárne na internú virtualizáciu; multitenancy a IaaS funkcionalita sú obmedzené.
- **Eucalyptus** [9]: Projekt je už dlhodobo neaktívny, vývoj stagnuje, webové stránky sú minimálne udržiavané a komunita takmer neexistuje. Z pohľadu udržateľnosti: platforma nemá perspektívu ďalšieho vývoja a nasadenie by bolo rizikové z hľadiska bezpečnosti, podpory aj kompatibility s modernými štandardmi.
- **Harvester** [10]**, KubeVirt** [11]**, Docker Swarm, Nomad, Rancher** - kontajnerovo orientované riešenia. Ich nasadenie by si vyžadovalo zásadnú zmenu architektúry projektu a neplnia IaaS požiadavky priamo.
- **Komerčné riešenia (VMware, Nutanix, Azure Stack, OpenShift Virtualization)** – Vylúčené z dôvodu licenčnej politiky, nákladov alebo závislosti na uzavretých ekosystémoch, ktoré nie sú vhodné pre výskumný open-source orientovaný projekt.

V ďalšej časti nasleduje detailná analýza vybraných troch riešení (OpenStack, CloudStack, Proxmox VE, Kubernetes) podľa sady stanovených hodnotiacich kritérií.

3 ANALÝZA KANDIDÁTOV NA CLOUDOVÚ/ VIRTUALIZAČNÚ PLATFORMU

3.1 FUNKČNÉ KRITÉRIÁ

3.1.1 K1: Podpora IaaS funkcionality (váha: 5)

Odôvodnenie: Projekt vyžaduje plnú funkcionalitu IaaS – správu virtuálnych strojov (VM), diskov, sietí, obrazov, snapshotov a samostatných zdrojov. Platforma musí podporovať vytváranie a správu rôznych typov inštancií, umožniť priradenie zdrojov podľa potreby tenantov a zároveň ponúknuť robustné API rozhranie pre programovateľný prístup. Schopnosť poskytovať kompletnú IaaS funkcionalitu je kľúčová pre zabezpečenie pružného a spoľahlivého prevádzkovania SOC ako služby.

Odporúčanie pre hodnotenie:

- 5 bodov: Podpora kompletných IaaS funkcií (VM, siete, disky, snapshoty), silné API rozhrania, potvrdené referencie.
- 3 body: Väčšina základných funkcií IaaS dostupná, API s určitými limitmi.
- 1 bod: Obmedzená funkcionalita IaaS, slabá alebo žiadna API podpora.

3.1.1.1 OpenStack

Popis funkcionality: OpenStack je v súčasnosti de-facto štandardom pre open-source IaaS cloud. Jeho architektúra je modulárna a zahŕňa špecializované komponenty pre rôzne IaaS funkcionality. OpenStack je globálne nasadený v stovkách produkčných prostredí, vrátane veľkých podnikov, telekomunikačných operátorov a výskumných inštitúcií. Jeho IaaS funkcionalita je overená v rozsiahlych a kritických nasadeniach.

- **VM (Compute – Nova):** Plná správa životného cyklu VM (spúšťanie, zastavovanie, pauzovanie, škálovanie, migrácia), podpora rôznych hypervisorov (KVM, vSphere, Hyper-V, Xen).
- **Disky (Block Storage – Cinder):** Dynamické vytváranie a pripájanie blokových úložísk k VM, podpora rôznych backendov (LVM, Ceph, NetApp, EMC, atď.), možnosť vytvárania snapshotov diskov.
- **Siete (Networking – Neutron):** Pokročilá sieťová virtualizácia, vrátane tvorby virtuálnych sietí, podsietí, smerovačov, bezpečnostných skupín (firewall na úrovni VM), plávajúcich IP adries a VPN. Podpora pre rôzne sieťové pluginy.
- **Obrazy (Image Service – Glance):** Centrálne úložisko pre obrazy VM (VM templates), umožňuje správu a verzionovanie.
- **Snapshoty (Nova, Cinder):** Podpora pre snapshoty VM (cez Nova) a diskov (cez Cinder) pre účely zálohovania a obnovy.
- **API:** Každý komponent OpenStacku poskytuje robustné a dobre zdokumentované RESTful API, ktoré umožňuje plnú programovateľnosť a integráciu s externými nástrojmi. Existuje aj rozsiahla podpora pre CLI a SDK.

Odôvodnenie: OpenStack je ucelené IaaS riešenie s plnou podporou správy VM (Nova), diskov a úložiska (Cinder, Swift), siete (Neutron), image služieb (Glance), snapshotov a manažmentu identít (Keystone) či s dodatočnými komponentmi pre orchestráciu a správu chýb, čo zabezpečuje vysokú dostupnosť [12]. Vďaka rozsiahlej modularite OpenStack vie ovládať veľké zdroje výpočtov, úložísk a sietí, ktorú je vďaka dashboardu či štandardizovanému REST API možné dynamicky škálovať a automatizovať, čo potvrdzujú aj lokálne nasadenia v akademickej sfére ([12] [13]). OpenStack je považovaný za de facto štandard pre open-source cloudy.

Hodnotenie: 5 b

3.1.1.2 CloudStack

Popis IaaS funkcionality: CloudStack je open-source IaaS platforma, ktorá sa zameriava na jednoduchosť použitia a škálovateľnosť [2]. CloudStack je taktiež využívaný v produkčných prostrediach, jeho ekosystém a komunita sú menšie ako pri OpenStacku. Je obľúbený pre svoju spoľahlivosť a jednoduchosť.

- **VM:** Ponúka komplexnú správu životného cyklu VM, podporuje rôzne hypervisory (KVM, VMware ESXi, XenServer, Hyper-V). Umožňuje definovanie a správu výpočtových a úložiskových "ponúk" (service offerings).
- **Disky:** Poskytuje správu dátových úložísk pre VM, vrátane perzistentných diskov, snapshotov diskov a šablón VM.
- **Siete:** Ponúka sieťovú virtualizáciu s podporou pre izolované virtuálne siete (VLAN, VxLAN), firewall pravidlá (security groups), IP adresy a load balancing. Sieťové funkcie sú integrované a jednoduchšie na konfiguráciu ako v Neutrone (OpenStack).
- **Obrazy:** Podporuje správu šablón VM (images) a ISO súborov.
- **Snapshoty:** Poskytuje možnosť vytvárať snapshoty VM a diskov.
- **API:** CloudStack disponuje robustným RESTful API, ktoré umožňuje plnú kontrolu nad IaaS zdrojmi a automatizáciu operácií.

Posúdenie: CloudStack ponúka integrovanú správu VM, sieťovania, úložiska, obrazov a snapshotov. Poskytuje jednoduchšie rozhranie (UI aj API) než OpenStack, pričom jeho funkcie sú stále vhodné aj pre enterprise nasadenia. Limitom môže byť menšia miera modularity, no v základných IaaS funkciách nijako výrazne nezaostáva. Má menšiu komunitu, menší ekosystém doplnkov a menšiu flexibilitu pre budúce rozšírenia čo je dané jeho architektúrou.

Hodnotenie: 4,5 b

3.1.1.3 Proxmox VE

Popis funkcionality: Proxmox VE je open-source virtualizačná platforma, ktorá sa primárne zameriava na správu virtuálnych strojov a kontajnerov, s integrovanými funkciami pre klastrovanie a úložisko. Nejde o plnohodnotnú cloudovú platformu v zmysle komplexného IaaS frameworku s natívnou podporou multitenancie, IaaS katalógov, pokročilých cloudových služieb a virtualizácie ako OpenStack. Veľmi populárny v SMB, akademickom a výskumnom prostredí pre jeho jednoduchosť, výkon a otvorenosť:

- **VM (KVM):** Plná správa VM (vytváranie, spúšťanie, migrácia, zálohovanie, obnova) a LXC. Podporuje rôzne operačné systémy a má dobrú správu zdrojov.
- **Disky:** Plná správa úložiska pre VM a LXC kontajnery, vrátane lokálnych (LVM, ZFS) a sieťových úložísk (NFS, SMB/CIFS, iSCSI, Ceph). Podporuje vytváranie a pripájanie virtuálnych diskov.
- **Siete:** Základná sieťová konfigurácia pre VM (mosty, VLAN tagging). Nemá pokročilé funkcie sieťovej virtualizácie ako virtuálne smerovače, bezpečnostné skupiny riadené na úrovni platformy alebo automatizovanú alokáciu IP adries pre tenantov. Tieto by si vyžadovali manuálnu konfiguráciu alebo externé nástroje.
- **Obrazy:** Podporuje správu ISO obrazov pre inštaláciu OS a šablón VM/LXC kontajnerov.
- **Snapshoty:** Plná podpora pre snapshoty VM a kontajnerov na úrovni úložiska.
- **API:** Proxmox VE má dobre zdokumentované RESTful API, ktoré umožňuje automatizáciu mnohých úloh správy VM, úložiska a klastra ako aj webové ovládanie. API je menej štandardizované a ponúka nižšie možnosti integrácie.

Posúdenie: Proxmox VE poskytuje funkcie pre správu VM, diskov, obrazov a snapshotov. Má robustné API rozhranie. Avšak, jeho "IaaS" funkcionalita je obmedzená v oblasti pokročilej sieťovej virtualizácie a natívnej podpory multitenancie na cloudovej úrovni (absencia napr. projektov, kvót, komplexných sieťových služieb pre tenantov ako je v OpenStacku/CloudStacku). Tieto chýbajúce "cloud-native" IaaS prvky by museli byť postavené nad Proxmox-om manuálne alebo pomocou externých nástrojov, čo znižuje jeho schopnosť poskytnúť *kompletnú* IaaS službu priamo z krabice pre multitenantné prostredie.

Hodnotenie: 3 b

3.1.1.4 Kubernetes

Popis IaaS funkcionality: Kubernetes je primárne platforma pre orchestráciu kontajnerov (PaaS – Platform as a Service) a správu mikroslužieb. Jeho hlavným cieľom je automatizácia nasadenia, škálovanie a správa kontajnerizovaných aplikácií. Nie je navrhnutý ako IaaS platforma pre virtuálne stroje a ich priamu správu. Dominantná platforma pre kontajnerizované aplikácie:

- **VM:** Kubernetes natívne nespravuje virtuálne stroje. Existujú však projekty ako **KubeVirt**, ktoré umožňujú spúšťať a spravovať VM ako kontajnery v rámci Kubernetes klastra. Toto však nie je natívna IaaS funkcionalita Kubernetes samotného.
- **Disky:** Kubernetes má robustnú správu perzistentného úložiska pre kontajnery (Persistent Volumes, Persistent Volume Claims), ale toto je určené pre kontajnery, nie pre tradičné blokové disky priamo pre VM (bez KubeVirt).
- **Siete:** Kubernetes poskytuje pokročilú sieťovú abstrakciu pre kontajnery (Service, Ingress, Network Policies), ale toto je odlišné od sieťovej virtualizácie pre VM v IaaS platformách. Pre VM by sa muselo spoliehať na underlying IaaS, alebo na komplexné KubeVirt riešenie.
- **Obrazy:** Spravuje kontajnerové obrazy, nie obrazy VM.
- **Snapshoty:** Podporuje snapshoty perzistentných zväzkov kontajnerov, nie VM.
- **API:** Kubernetes má výkonné a flexibilné API pre orchestráciu kontajnerov. Avšak, pre IaaS funkcie pre VM by sa muselo použiť KubeVirt API, ktoré je nadstavbou.

Odôvodnenie: Kubernetes nie je IaaS platforma. Jeho schopnosť spravovať VM je závislá od doplnkov ako KubeVirt, čo nie je natívna funkcionalita a prináša výraznú komplexnosť. Nesplňuje požiadavku na "kompletné IaaS funkcie" pre VM priamo z krabice. Hoci je kľúčový pre moderné aplikácie, pre priame hostovanie SOCaaS na úrovni IaaS (ako VM) je nevhodný bez značných dodatočných úprav a integrácií. Kubernetes je považovaný skôr za PaaS alebo CaaS (Container as a Service), nie za IaaS, a vyžaduje IaaS vrstvu pod sebou na poskytovanie zdrojov.

Hodnotenie: 1 b

3.1.1.5 Zhodnotenie

Kritérium K1 vyžaduje plnú funkcionalitu IaaS, vrátane správy virtuálnych strojov (VM), diskov, sietí, obrazov, snapshotov a samostatných zdrojov, s robustným API a potvrdenými referenciami. Skóre sú nasledovné:

- **OpenStack**: 5 bodov – Poskytuje kompletné IaaS funkcie, vrátane VM, sietí, diskov a snapshotov, s silným API a širokou adopciou.
- **CloudStack**: 4,5 bodov – Podobne ako OpenStack, ponúka plnú IaaS funkcionalitu s API a referenciami.
- **Proxmox**: 3 body – Podporuje VM, disky, siete, snapshoty a má API, s potvrdenými referenciami ako IaaS platforma.
- **Kubernetes**: 1 bod – Nie je IaaS platforma, slúži na správu kontajnerov, nie na poskytovanie infraštruktúry.

3.1.2 K2: Modularita a architektúra (váha: 4)

Odôvodnenie: Modularita architektúry zásadne ovplyvňuje rozšíriteľnosť a škálovateľnosť riešenia. Platforma, ktorá je rozdelená do samostatných komponentov (napr. Nova – compute, Neutron – networking, Cinder – block storage), umožňuje nezávislé nasadzovanie, ladenie a výmenu modulov podľa potrieb. Taktiež je výhodné, ak platforma podporuje modernú architektúru mikroslužieb, pluginy, kontajnerizáciu a orchestračné vrstvy. Vyššia modularita zároveň znižuje riziko výpadku celého systému pri poruche jednej služby.

Odporúčanie pre hodnotenie:

- **5 bodov:** Vysoká modularita, mikroslužby, samostatne nasaditeľné komponenty, jasná integrácia pluginov.
- **3 body:** Modulárnosť je čiastočná, niektoré komponenty sú navzájom závislé.

- **1 bod:** Monolitická architektúra, obmedzené možnosti úprav alebo výmeny komponentov.

3.1.2.1 OpenStack

Popis: OpenStack je navrhnutý ako súbor vzájomne interagujúcich, ale samostatných projektov (komponentov), ktoré poskytujú špecifickú službu. Každý hlavný komponent (napr. Nova pre výpočty, Neutron pre siete, Cinder pre blokové úložiská, Swift pre objektové úložiská, Glance pre obrazy, Keystone pre identitu) má vlastné API a môže byť nasadený, škálovaný a udržiavaný nezávisle od ostatných. Každý komponent beží ako oddelený démon alebo služba a spolu komunikujú cez definované API rozhrania a fronty správ, čím vytvárajú jednotný cloud systém. Táto mikroservisná filozofia umožňuje nasadiť len potrebné komponenty a podľa potreby ich nahrádzať či škálovať. OpenStack tiež podporuje pluginovú architektúru v rámci jednotlivých služieb – napríklad Neutron umožňuje rôzne sieťové ovládače (ML2 pluginy), Cinder má ovládače pre rôzne storage platformy, atď. Vďaka otvorenosti kódu je systém vysoko rozšíriteľný; komunita pridáva nové projekty (napr. Magnum pre kontajnery, Ironic pre bare-metal, Heat pre orchestráciu) bez nutnosti meniť základnú architektúru. Architektúra teda poskytuje extrémnu flexibilitu a prispôsobiteľnosť – organizácia si môže vybrať a nakonfigurovať len podmnožinu služieb podľa svojich potrieb. Táto decentralizovaná architektúra výrazne zvyšuje odolnosť a flexibilitu systému [15] [16]. OpenStack neustále adaptuje moderné prístupy, vrátane kontajnerizácie svojich riadiacich služieb (napr. cez Kolla-Ansible alebo OpenStack-Helm), čím ďalej zvyšuje modularitu a zjednodušuje správu [17]. OpenStack podporuje:

- **Modularitu**: Je rozdelený na samostatné služby, ako Nova (výpočty), Neutron (siete), Cinder (úložisko), ktoré môžu byť nasadené nezávisle. Tento dizajn umožňuje flexibilitu a škálovateľnosť, čo je kľúčové pre projekt INNOSOC.
- **Mikroslužby**: OpenStack používa architektúru mikroslužieb, kde každá služba je samostatnou aplikáciou s vlastným životným cyklom, čo znižuje riziko výpadku celého systému.
- **Samostatne nasaditeľné komponenty**: Áno, každá služba môže byť nasadená a spravovaná samostatne, čo umožňuje nezávislé ladenie a výmenu modulov.
- **Integrácia pluginov**: Podporuje širokú škálu pluginov a integrácií, napríklad pre hypervizory, úložisko a siete, čo zvyšuje prispôsobiteľnosť.
- **Kontajnerizácia a orchestrácia**: Prostredníctvom komponentu Magnum podporuje kontajnery a orchestráciu s Kubernetes, čo je v súlade s požiadavkami projektu.

Odôvodnenie: OpenStack je navrhnutý ako plne modulárny systém. Jeho komponenty sú samostatné mikroslužby s vlastnými API, čo umožňuje flexibilné nasadenie, škálovanie a odolnosť voči zlyhaniu. Podpora pluginov (najmä v Neutrone) a trend kontajnerizácie riadiacich služieb reflektuje na potreby moderných riešení. **Nevýhodou** je vysoká komplexita: modulárnosť so sebou prináša náročnejšie nasadenie a správu. Celkovo však architektúra poskytuje maximálnu flexibilitu a prispôsobiteľnosť.

Hodnotenie: 4,5 b

3.1.2.2 CloudStack

Popis: CloudStack má tiež modulárnu architektúru, hoci je často v literatúre označovaná ako "jednoduchšia" alebo "monolitickejšia" v porovnaní s OpenStackom [18]. Hlavnými komponentmi sú CloudStack Management Server, ktorý riadi celý cloud, a CloudStack Agents, ktoré bežia na hypervisoroch. Napriek tomu sú služby ako výpočet, sieť a úložisko logicky oddelené a riadené manažment serverom. CloudStack podporuje pluginy pre hypervisory (VMware, KVM, XenServer), úložiská a sieťové služby, čo umožňuje určitú

mieru prispôsobenia a rozšíriteľnosti [18]. Nie je však primárne postavený na architektúre mikroslužieb v takom rozsahu ako OpenStack a komponenty systému nie sú tak prísne oddelené ako v OpenStacku. Porovnávacie štúdie medzi OpenStackom a CloudStackom často poukazujú na to, že zatiaľ čo CloudStack je jednoduchší na nasadenie, jeho modularita je menej granulárna ako pri OpenStacku. Vďaka integrovanej architektúre sa CloudStack inštaluje ako jeden produkt, čo značne zjednodušuje nasadenie a upgrady – systém sa aktualizuje centrálne a všetky časti sú navrhnuté na vzájomnú kompatibilitu. CloudStack poskytuje:

- **Modularitu**: Má modulárnu architektúru s komponentmi ako Management Server, Agent a pluginmi pre hypervizory a úložisko. Nie je však tak striktne založený na mikroslužbách ako OpenStack, čo môže obmedziť jeho flexibilitu.
- **Mikroslužby**: Podporuje niektoré aspekty mikroslužieb, ale nie tak rozsiahle, s väčšinou funkcií integrovaných do Management Servera.
- **Samostatne nasaditeľné komponenty**: Väčšina komponentov môže byť nasadená samostatne, ale existujú určité závislosti, napríklad medzi Management Serverom a Agentmi, čo môže zvýšiť riziko výpadku.
- **Integrácia pluginov**: Podporuje pluginy pre hypervizory (napr. VMware, KVM) a úložisko, ale je menej flexibilný ako OpenStack.
- **Kontajnerizácia a orchestrácia**: Má obmedzenú natívnu podporu pre kontajnery, hoci môže byť integrovaný s externými orchestrátormi ako Kubernetes.

Odôvodnenie: CloudStack ponúka jednoduchú, integrovanú architektúru s podporou multi-tenancy, čo je plus pre spoľahlivosť a ľahkú správu v bezpečnostne orientovanom cloude. Zásluhou menšej komplexity je nižšia pravdepodobnosť konfiguračných chýb či problémov pri upgrade – to prispieva k bezpečnosti aj dostupnosti. Bodovo však zaostáva v modularite: monolitický dizajn neumožňuje tak hlboké prispôsobenie alebo výmenu komponentov ako OpenStack. Rozširovanie je limitované na definované plugin rozhrania a celková flexibilita architektúry je menšia. Celkovo je CloudStack vhodný pre stredne veľké viacnájomnícke cloudy, kde sa vyžaduje robustná funkcionalita bez komplikovanej správy, avšak pokročilí používatelia môžu narážať na hranice možností architektúry pri veľmi špecifických potrebách..

Hodnotenie: 3 b – Čiastočná modularita

3.1.2.3 Proxmox VE

Popis: Proxmox VE je postavený v porovnaní s cloudovými platformami na monolitickejšej architektúre. Je to integrované riešenie pre správu virtualizácie (KVM), kontajnerov (LXC), úložiská (Ceph, ZFS) a sieťovanie s vlastným webovým rozhraním, ktoré spravuje základné služby (virtuálne stroje, úložisko, sieť). Klastrovacie funkcie sú vstavané. Hoci používa komponenty s otvoreným zdrojovým kódom (KVM, ZFS, Corosync pre klaster), nie sú primárne navrhnuté ako samostatné služby s vlastnými verejnými API, ktoré by umožňovali ich nezávislé nasadzovanie alebo výmenu. Rozšíriteľnosť je skôr prostredníctvom integrácie backendov úložiska alebo pomocou skriptov a priameho prístupu k Linuxovým súčastiam [19]. Poskytuje

- **Modularita**: Je založený na monolitickom dizajne, kde veľa funkcií je integrovaných do jediného uzla, hoci podporuje clusterizáciu pre vysokú dostupnosť. Tento dizajn obmedzuje rozšíriteľnosť a škálovateľnosť.
- **Mikroslužby**: Nepodporuje architektúru mikroslužieb; je viac integrovaná platforma, čo zvyšuje riziko výpadku celého systému pri poruche uzla.
- **Samostatne nasaditeľné komponenty**: Nie, veľa funkcií je viazaných na uzol, čo obmedzuje nezávislé nasadenie a ladenie.
- **Integrácia pluginov**: Podporuje niektoré pluginy, ako softvérovo definované úložisko a siete, ale nie je tak flexibilný ako OpenStack alebo Kubernetes [21].

- **Kontajnerizácia a orchestrácia**: Podporuje kontajnery prostredníctvom LXC, ale nemá natívnu orchestráciu ako Kubernetes, čo obmedzuje jeho použitie pre moderné mikroslužby.

Odôvodnenie: Z pohľadu kritéria K2 má Proxmox najmenej modulárnu architektúru spomedzi porovnávaných platforiem. Jeho dizajn „všetko v jednom“ prináša síce jednoduchosť nasadenia a nízku komplexitu (čo je pozitívne z hľadiska správy menších nasadení), no chýba mu výraznejšia rozčleniteľnosť komponentov a podpora viacnájomníckeho modelu. Pre SOCaaS so striktnou segregáciou nájomcov by Proxmox musel byť doplnený o vlastné nadstavby; samotná architektúra na také použitie nebola primárne navrhnutá. Výhodou je, že integrácia výpočtu, sietí a úložiska v Proxmoxe je veľmi tesná a optimalizovaná – pre menšie škálovateľné celky to prináša efektivitu.

Hodnotenie: 1 b – monolitická architektúra

3.1.2.4 Kubernetes

Popis: Kubernetes je navrhnutý s podporou vysokej modularity a je postavený na architektúre mikroslužieb. Jeho riadiaca rovina (Control Plane) sa skladá z komponentov ako API Server, etcd, Scheduler a Controller Manager, z ktorých každý beží ako samostatná služba. Pracovné uzly (Worker Nodes) spúšťajú kubelet a kube-proxy. Kubernetes je rozsiahle rozšíriteľný prostredníctvom Custom Resource Definitions (CRD), operátorov a rozsiahleho ekosystému pluginov (napr. CNI pre sieťovanie, CSI pre úložisko) [5]. Táto architektúra je inherentne distribuovaná a odolná [20]. Poskyttuje:

- **Modularita**: Je vysoko modulárny, s komponentmi ako API Server, Controller Manager, Scheduler, ktoré môžu byť nasadené samostatne. Tento dizajn podporuje rozšíriteľnosť a škálovateľnosť, čo je kľúčové pre projekt INNOSOC.
- **Mikroslužby**: Je navrhnutý pre orchestráciu mikroslužieb v kontajneroch, čo znižuje riziko výpadku pri poruche jednej služby.
- **Samostatne nasaditeľné komponenty**: Áno, každý komponent môže byť nasadený a spravovaný samostatne, čo umožňuje nezávislé ladenie a výmenu modulov.
- **Integrácia pluginov**: Podporuje širokú škálu pluginov prostredníctvom rozhraní ako CNI (Container Network Interface), CSI (Container Storage Interface), čo zvyšuje prispôsobiteľnosť.
- **Kontajnerizácia a orchestrácia**: Je špičkovým orchestrátorom kontajnerov, podporujúcim škálovanie, rolling updates a iné moderné funkcie, čo je v súlade s požiadavkami projektu.

Odôvodnenie: Kubernetes je príkladom modernej a vysoko modulárnej architektúry založenej na mikroslužbách. Jeho komponenty sú samostatné, distribuované a rozsiahle rozšíriteľné prostredníctvom CRD a pluginov. Hoci nie je primárne IaaS platformou pre VM, samotná architektúra je špičkovo modulárna a odolná, čo je v súlade s požiadavkami kritéria. Veľkým prínosom je zdieľanie ekosystému s cloud-native technológiami – bezpečnostné monitoring nástroje, CI/CD pipeline, logovanie, to všetko môže byť unifikované pre kontajnery aj VM. Hoci multi-tenantnosť nie je úplne bez námahy, platforma poskytuje dostatočné prostriedky ju dosiahnuť. V porovnaní s tradičnými cloud platformami tu vidíme nižšiu vertikálnu komplexitu (menej rôznorodých služieb než OpenStack) a lepšiu horizontálnu škálovateľnosť (ľahké pridanie nodov, kontrolérov). Za kritérium K2 získava Kubernetes+KubeVirt najvyššie hodnotenie, pretože spĺňa mikroservisnú modularitu, rozšíriteľnosť aj perspektívnosť architektúry pre budúce potreby. Z pohľadu SOCaaS to znamená platformu, na ktorej možno stavať multi-tenantné bezpečnostné služby s využitím najnovších trendov v automatizácii a orchestrácii

Hodnotenie: 5 b

3.1.3 K3: Podpora orchestrácie a automatizácie (váha: 4)

Odôvodnenie: Platforma by mala podporovať nástroje ako Heat (OpenStack orchestration), Terraform (infrastructure as code), Ansible (automatizácia) alebo Kubernetes Operator pre efektívnu správu a nasadzovanie komplexných infraštruktúrnych scénarov. Táto schopnosť umožní rýchle škálovanie služieb, automatizované rozbaľovanie prostredí, opakovateľné testovanie aj zotavenie pri výpadkoch. V prostredí SOC, kde sa často menia požiadavky a topológia siete, je orchestrácia nevyhnutná pre zabezpečenie flexibility, udržateľnosti a prevádzkovej efektivity.

Odporúčanie pre hodnotenie:

- 5 bodov: Výborná podpora (Heat, Terraform, Ansible, Kubernetes), automatizácia robustná a flexibilná.
- 3 body: Základná orchestrácia dostupná, čiastočné limity v podpore nástrojov.
- 1 bod: Veľmi obmedzená alebo žiadna automatizácia, slabá integrácia s populárnymi nástrojmi.

3.1.3.1 OpenStack

Popis: OpenStack je navrhnutý s ohľadom na automatizáciu a orchestráciu, čo je kľúčové pre budovanie dynamických cloudových prostredí:

- **Heat:** Je natívny orchestrátor OpenStacku, ktorý umožňuje definovať komplexné cloudové prostredia ako kód (Infrastructure as Code – IaC) pomocou Heat Orchestration Templates (HOT) [21]. Podporuje nasadenie a správu celých "stackov" aplikácií, vrátane VM, sietí, úložísk, bezpečnostných skupín a auto-škálovacích skupín.
- **Terraform:** OpenStack má oficiálneho poskytovateľa (providera) pre HashiCorp Terraform [22], čo umožňuje spravovať zdroje OpenStacku pomocou deklaratívneho kódu Terraformu. Je to veľmi populárny nástroj pre multi-cloud a hybrid-cloud IaC. Plne kompatibilný, umožňuje pokročilú automatizáciu.
- **Ansible:** OpenStack má rozsiahle moduly Ansible pre automatizáciu správcovských úloh a nasadenia aplikácií na VM v OpenStacku [23]. Mnoho nasadení OpenStacku (napr. Kolla-Ansible) samo využíva Ansible na inštaláciu a konfiguráciu.
- **Kubernetes Operators:** Pre riadenie OpenStacku bežiaceho na Kubernetes (napr. cez OpenStack-Helm alebo iné projekty) je možné využiť Kubernetes Operators na automatizované nasadenie a správu komponentov OpenStacku.
- **Ceilometer/AODH:** OpenStack má aj vstavané mechanizmy pre monitoring (Ceilometer) a auto-škálovanie (AODH - Alarm Orchestration and Decision Engine), ktoré umožňujú automatické prispôsobovanie zdrojov na základe záťaže.

Odôvodnenie: OpenStack je postavený na API-first prístupe, čo z neho robí ideálnu platformu pre automatizáciu. S natívnym orchestrátorom Heat, robustnou integráciou s Terraformom a rozsiahlymi modulmi Ansible ponúka výbornú podporu pre IaC a automatizáciu komplexných scenárov, čo je pre dynamické prostredie SOCaaS kľúčové.

Hodnotenie: 5 b

3.1.3.2 CloudStack

Popis: CloudStack tiež kladie dôraz na automatizáciu a má dobre definované API.

- **API:** CloudStack poskytuje rozsiahle RESTful API, ktoré umožňuje programovo spravovať všetky aspekty IaaS (VM, siete, úložiská, atď.) [24]. To je základ pre ďalšiu automatizáciu.

- **Terraform:** Existuje oficiálny provider pre Terraform pre CloudStack [25], čo umožňuje definovať a spravovať CloudStack infraštruktúru pomocou IaC.
- **Ansible:** Podobne ako OpenStack, aj CloudStack má podporu prostredníctvom modulov Ansible, ktoré umožňujú automatizáciu úloh správy CloudStacku a nasadzovania aplikácií [26]. Dostupné moduly sú menej rozvinuté.
- **CloudStack Orchestration:** CloudStack nemá priamo ekvivalent Heat Orchestration ako OpenStack, ale jeho Service Offerings a Network Offerings spolu s API umožňujú určitú mieru orchestrácie. Zvyčajne sa pre komplexnejšie scenáre spolieha na externé nástroje (ako Terraform alebo vlastné skripty/aplikácie volajúce API).

Odôvodnenie: CloudStack poskytuje robustné API a dobrú integráciu s populárnymi IaC a automatizačnými nástrojmi ako Terraform a Ansible. Hoci mu chýba natívny komplexný orchestrátor typu Heat, jeho schopnosť programovej správy zdrojov prostredníctvom API je možná.

Hodnotenie: 4 b

3.1.3.3 Proxmox VE

Popis: Proxmox VE ponúka API a nástroje na automatizáciu, ale je primárne navrhnutý pre správu virtualizácie na úrovni serverov, nie ako plnohodnotná cloudová orchestrácia:

- **REST API:** Proxmox VE má dobre zdokumentované RESTful API, ktoré umožňuje automatizovať väčšinu úloh správy VM, kontajnerov, úložiska a klastra [27]. To je základ pre skriptovanie a integráciu.
- **Terraform:** Existuje neoficiálny, ale populárny Terraform provider pre Proxmox VE [28], ktorý umožňuje spravovať VM, úložiská a siete pomocou IaC.
- **Ansible:** Proxmox má rozsiahlu podporu prostredníctvom modulov Ansible [29], ktoré umožňujú automatizovať nasadenie a konfiguráciu VM, ako aj správu Proxmox klastra. Dostupné moduly sú menej rozvinuté.
- **Chýbajúce natívne cloudové orchestrácie:** Proxmox nemá vstavaný orchestrátor pre komplexné cloudové scenáre (napr. automatické škálovanie aplikácií, správu "stackov" viacerých prepojených komponentov s komplexnými závislosťami) ako Heat v OpenStacku. Takéto funkcie by si vyžadovali implementáciu cez externé nástroje a skriptovanie na báze jeho API.

Odôvodnenie: Proxmox VE má robustné API a veľmi dobrú integráciu s populárnymi automatizačnými nástrojmi ako Terraform a Ansible pre správu virtuálnych zdrojov. Chýba mu však natívny orchestrátor pre komplexné cloudové scenáre a automatické škálovanie služieb v multitenantnom prostredí. Orchestrácia by bola možná, ale vyžadovala by si významné dodatočné úsilie prostredníctvom externých nástrojov a skriptovania.

Hodnotenie: 3 b

3.1.3.4 Kubernetes

Popis: Kubernetes je veľmi silný v orchestrácii a automatizácii, hoci primárne pre kontajnery:

- **Natívna Orchestrácia:** Samotný Kubernetes je orchestrátor pre kontajnery. Spravuje nasadenie, škálovanie, samoopravu a riadenie životného cyklu kontajnerizovaných aplikácií. Koncepty ako Deployments, StatefulSets, Services, Horizontal Pod Autoscalers sú jeho jadrom [30].

- **Kubernetes Operators:** Kľúčový mechanizmus pre automatizáciu komplexných aplikácií. Operátori sú softvéroví agenti, ktorí automatizujú nasadenie a správu aplikácií v Kubernetes, čo je ideálne pre SOCaaS služby [31].
- **Infrastructure as Code (IaC):** Všetko v Kubernetes je definované deklaratívne pomocou YAML súborov, čo je prirodzená forma IaC.
- **Terraform:** Kubernetes má oficiálny provider pre Terram [32], ktorý umožňuje spravovať zdroje Kubernetes pomocou Terraformu.
- **Ansible:** K8s má tiež bohatú podporu prostredníctvom modulov Ansible pre interakciu s klastrom Kubernetes a správu jeho zdrojov [33].
- **KubeVirt:** Pre automatizáciu a orchestráciu VM na Kubernetes (ako kontajnerov) sa používa KubeVirt, ktorý využíva orchestráciu Kubernetes.

Odôvodnenie: Kubernetes je špičková platforma pre orchestráciu a automatizáciu kontajnerizovaných záťaží. Hoci jeho primárnym zameraním nie sú VM, jeho základné princípy (deklaratívna konfigurácia, operátori, IaC) a rozsiahla podpora nástrojov ako Terraform a Ansible z neho robia extrémne flexibilnú platformu pre automatizáciu komplexných scénarov, dokonca aj pre VM prostredníctvom KubeVirt.

Hodnotenie: 5 b

3.2 TECHNICKÉ KRITÉRIÁ

3.2.1 K4: Výkonnosť a škálovateľnosť (váha: 5)

Odôvodnenie: Hodnotí sa, ako efektívne dokáže riešenie využívať zdroje, aké má limity z hľadiska počtu VM, kontajnerov či tenantov, a aké mechanizmy automatického škálovania podporuje. Dôležitá je aj latencia a celkový výkon pri spracovaní údajov, najmä v reálnom čase pre účely monitoringu bezpečnostných udalostí. Riešenie musí byť schopné rásť s počtom VM, tenantov a služieb. Hodnotí sa horizontálna a vertikálna škálovateľnosť. V podmienkach projektu potrebujeme platformu, ktorá dokáže rásť spolu s počtom klientov SOC služby a objemom spracovávaných dát.

Preto tomuto kritériu prislúcha **váha 5** – maximálna dôležitosť, keďže nedostatočná výkonnosť alebo obmedzené škálovanie by predstavovali zásadné riziko.

Odporúčanie pre hodnotenie:

- 5 bodov: Vynikajúca škálovateľnosť, potvrdená schopnosť obslúžiť stovky VM, horizontálne a vertikálne škálovanie.
- 3 body: Primeraná škálovateľnosť s istými obmedzeniami (desiatky VM bez výrazného poklesu výkonu).
- 1 bod: Slabý výkon, výrazné škálovacie limity už pri nízkom počte VM.

3.2.1.1 OpenStack

Popis: OpenStack je vysoko škálovateľné riešenie navrhnuté pre veľké nasadenia, zvládajúce jednotky až tisíce virtuálnych strojov. Podporuje horizontálne aj vertikálne škálovanie komponentov (Nova, Neutron, Cinder):

- Nova plánovač dokáže efektívne rozkladať záťaž na stovky výpočtových uzlov a systém má zabudované mechanizmy na automatické vyvažovanie záťaže a migráciu VM pri problémoch výkonu.
- **Horizontálna škálovateľnosť:** Každý komponent OpenStacku (Nova, Neutron, Cinder, atď.) môže byť škálovaný horizontálne pridaním ďalších inštancií služby na

Compute nodoch. To umožňuje obsluhovať stovky až tisíce hypervizorov a tisíce virtuálnych strojov [36] [37].

- **Vertikálna škálovateľnosť:** Podpora pre rôzne druhy VM ("flavors") umožňuje pridelenie väčšieho množstva CPU, RAM a diskového priestoru jednotlivým VM.
- **Horizontálne vs. vertikálne škálovanie:** OpenStack primárne škáluje horizontálne – pridaním ďalších výpočtových uzlov do klastru (prípadne rozdelením do viacerých „celkov“ alebo regionálnych klastrov pri obrovských nasadeniach). Zároveň umožňuje aj **vertikálne škálovanie VM** – administrátor alebo samotná platforma môže upraviť flavor (veľkosť) bežiacej VM. Podpora **hot-plug** pridávania CPU a RAM bola historicky limitovaná, no pri použití KVM a moderných OS je možné v OpenStacku dynamicky pridať vCPU či pamäť do bežiacej VM (vyžaduje to podporu v hypervízore aj vo vnútri hosťa). OpenStack tak vie reagovať aj na meniace sa nároky aplikácií bez nutnosti reštartu (napr. zväčšením RAM databázového servera za behu).
- **Využitie zdrojov:** OpenStack sa snaží o efektívne využitie podkladových fyzických zdrojov prostredníctvom pokročilého plánovania (scheduling) VM na základe dostupnosti zdrojov a definovaných politík.
- **Latencia a real-time**: Pre nízku latenciu a near real-time spracovanie dát (dôležité pre SOC) ponúka OpenStack možnosti ako SR-IOV a DPDK pre sieťovanie, čo umožňuje VM priamy prístup k hardvéru a obchádzanie virtuálnej sieťovej vrstvy hypervizora [3]. To výrazne znižuje latenciu pri spracovaní vysokého objemu sieťovej prevádzky.
- **Automatické škálovanie:** Pomocou Heat (orchestrácia) a Ceilometer/AODH (monitoring a alarmy) je možné implementovať automatické škálovanie VM na základe metrík zaťaženia či definovaných politík, čo zaisťuje dynamickú reakciu na zmeny záťaže.
- **Výkonnosť správy VM:** V OpenStacku beží každá VM priamo na hypervízore (typicky KVM) s takmer natívnym výkonom. Režijná záťaž OpenStacku sa prejavuje skôr na riadiacej rovine (API, plánovač, sieťový a úložný servis). Pri veľkých počtoch VM je dôležitá distribúcia záťaže – OpenStack umožňuje škálovať kontrolné služby nasadením viacerých inštancií (napr. paralelné API a scheduler uzly). Testy ukázali, že napríklad službu **Nova Scheduler** je vhodné spustiť vo viacerých vláknach/procesoch pri stovkách uzlov (v jednom teste bežalo 8 scheduler procesov pre 1000 uzlov). Sieťová služba Neutron podporuje distribuovaný routing (DVR) pre zníženie latencie a záťaže pri sieťovej komunikácii naprieč uzlami.
- **Latencia a limity:** Čo sa týka latencie, pri správnej konfigurácii si OpenStack uchováva rozumné časy vytvárania VM aj pri väčšom rozsahu. Napríklad Red Hat uvádza, že vylepšením výkonu databázy a schedulera sa darí udržať akceptovateľné časy spúšťania VM aj pri ~1000 uzloch. Každopádne pri extrémnom scale (stovky uzlov) je nutné počítať s určitým oneskorením niektorých operácií (napr. inicializácia siete pre VM môže trvať pár sekúnd). Typické limitácie sú dané konfiguráciou: predvolené nastavenia (napr. počet vláken Heat engine, veľkosť fronty v RabbitMQ) môžu byť nutné zväčšiť pre bezproblémový chod vo veľkých klastroch. Oficiálna dokumentácia odporúča pri ~1000 uzloch vyhradiť samostatné kontrolné uzly pre kľúčové služby (Nova, Neutron, Keystone atď.), aby sa predišlo úzkym hrdlám.

Podľa testov z praxe a akademických zdrojov (IEEE) je OpenStack schopný bez problémov škálovať do stoviek až tisícov VM. OS poskytuje

- Škálovateľnosť: Podporuje horizontálne škálovanie pridaním viac uzlov a vertikálne škálovanie zvýšením zdrojov na existujúcich uzloch. Dokumentácia naznačuje, že môže spravovať tisíce VM a výpočtových uzlov, s testami na 1 000 uzloch a 10-30 tisíc VM [36].

- Výkon: Optimalizovaný pre výkon, s funkciami ako živá migrácia, vyvažovanie záťaže a efektívne prideľovanie zdrojov. Je vhodný pre reálne záťaže, vrátane monitorovania bezpečnostných udalostí, s nízkou latenciou a vysokou priepustnosťou.
- Reálne použitie: Používaný veľkými organizáciami, ako telekomunikačné spoločnosti, vlády a podniky, čo dokazuje jeho schopnosť zvládať veľké infraštruktúry.

Odôvodnenie: OpenStack je navrhnutý pre extrémne vysokú škálovateľnosť, horizontálne aj vertikálne. Jeho modulárna architektúra umožňuje škálovanie jednotlivých služieb, a pokročilé sieťové techniky zabezpečujú nízku latenciu. Je preukázateľne schopný obslúžiť stovky až tisíce VM v produkčných prostrediach, čo plne spĺňa požiadavky na rast SOCaaS.

Hodnotenie: 5 b – Vynikajúca škálovateľnosť, potvrdená schopnosť obslúžiť stovky VM, s podporou horizontálneho a vertikálneho škálovania.

3.2.1.2 CloudStack

Popis: CloudStack je tiež známy svojou robustnou škálovateľnosťou a je optimalizovaný pre správu veľkých počtov VM. Výhodou CloudStacku je, že jednoduchosť architektúry prináša spočiatku nízku réžiu (rýchla inštancia cloudu pre menšie nasadenia), avšak pri extrémnej škále môže monolit naraziť na strop výkonu – vtedy je nutné škálovať vertikálne (výkonnejší HW pre management a DB, prípadne ich klastrovať). Výsledkom je, že CloudStack zvláda bežné veľké nasadenia výborne, no nad určitú hranicu (rádovo tisíce hostov) by mohol vyžadovať architektonické úpravy či rozdelenie medzi viacero nezávislých CloudStack inštancií:

- **Škálovateľnosť** CloudStacku je overená v mnohých produkčných nasadeniach poskytovateľov hostingu a cloudov; hoci nie je tak často nasadzovaný v úplne masívnych cloudoch ako OpenStack, dokáže obslúžiť stovky až nízke tisíce fyzických hostiteľov. CloudStack využíva zónovú architektúru – infraštruktúra sa delí na zóny, podsiete (pody) a klastre. Každý klaster zvyčajne zoskupuje niekoľko hostiteľov (hypervízorov) s spoločným úložiskom. Škálovanie prebieha pridávaním ďalších klastrov a zón, ktoré stále riadi centrálny management server (prípadne viacero v HA móde). Výhodou je, že záťaž sa tým do určitej miery rozloží (napr. databázovo zaťaženie podľa zón).
- **Horizontálna škálovateľnosť:** Architektúra CloudStacku je navrhnutá pre jednoduché horizontálne škálovanie pridaním ďalších výpočtových uzlov (hypervizorov) [38]. Jeden manažment server alebo klaster manažment serverov dokáže spravovať tisíce fyzických hostiteľov a desiatky tisíc VM. Dokáže spravovať veľké množstvo VM, ale zdá sa, že je menej škálovateľný ako OpenStack, s väčšou vhodnosťou pre menšie až stredné nasadenia.
- **Vertikálna škálovateľnosť:** Podporuje dynamické zmeny zdrojov (CPU, RAM) pre VM za behu, ak to hypervisor umožňuje. Administrátor môže upraviť “Service Offering” (t.j. parametre CPU/RAM) priradené VM – ak má VM nainštalované potrebné ovládače (napr. VMware Tools pri VMware, Xentools pri XenServeri), CloudStack vie za behu zvýšiť vCPU alebo RAM danej VM. Toto dynamické škálovanie CPU/RAM je podporované napr. pre VMware a XenServer hypervízory. Pri KVM platforme je zmena offeringu zvyčajne možná až po vypnutí VM (CloudStack reštartuje VM s novými parametrami). Hoci teda CloudStack nemá úplne univerzálny hot-plug pre všetky hypervízory, v prostredí s VMware/Xen dokáže vertikálne škálovať VM bez výpadku.
- **Výkon**: Dobrý výkon pre svoje použitie, ale nie tak optimalizovaný pre veľmi veľké alebo vysokovýkonné prostredia. Výskum pokrýva parametre ako CPU, pamäť, disk I/O a sieťový výkon, ale s menšími referenciami pre veľké škály [39].

- **Využitie zdrojov:** Efektívne plánovanie a prideľovanie zdrojov, s možnosťou nastavenia kvót pre používateľov a domény.
- **Latencia a real-time:** Závisí primárne od podkladového hypervizora a sieťovej infraštruktúry. CloudStack priamo neponúka také pokročilé optimalizácie sieťovej cesty ako SR-IOV v OpenStacku ako vstavanú cloudovú funkciu, ale umožňuje ich konfiguráciu na úrovni hypervizora.
- **Automatické škálovanie:** CloudStack má vstavané funkcie automatického škálovania (AutoScale), ktoré umožňujú pridávať alebo odoberať VM na základe definovaných metrík [5].
- **Reálne použitie:** Používaný viacerými poskytovateľmi cloudov a podnikmi, ale jeho adopcia nie je taká široká ako OpenStack, čo môže naznačovať obmedzenia v škálovateľnosti pre najväčšie nasadenia.

Odôvodnenie: Apache CloudStack ponúka veľmi dobrú výkonnosť a škálovateľnosť pre väčšinu typických nasadení. Jeho integrovaná povaha zjednodušuje správu a dosahuje efektivitu pri malých a stredných clusteroch, pričom bez problémov zvláda stovky uzlov a tisíce VM. Za najväčšie obmedzenie možno považovať centrálny management – pri extrémne veľkej infraštruktúre sa stáva úzkym hrdlom, čo znižuje pružnosť škálovania v porovnaní s distribuovanejšou architektúrou OpenStacku. OpenStack je preto často preferovaný v prostredí, kde sa plánuje škálovať do tisícov uzlov, hoci CloudStack vylepšeniami (caching, optimalizácie schedulera) dokázal výrazne posunúť svoj limit a spoľahlivo funguje aj v pomerne veľkých cloudoch.

Hodnotenie: 4 b

3.2.1.3 Proxmox VE

Popis: Proxmox VE je škálovateľný, ale primárne v rámci klastra a pre priamu správu VM:

- **Výkonnosť:** Z hľadiska výkonnosti jednej VM Proxmox využíva osvedčené technológie – hypervízor KVM pre plnú virtualizáciu a LXC pre kontajnery – takže výkon bežiacich VM je takmer natívny (KVM je typu 1 s minimálnou réžiou). Proxmox pridáva tenkú vrstvu managementu (webové rozhranie, API), ktorá sama o sebe výrazný výkonový overhead nepredstavuje. Znamená to, že pri porovnaní s inými platformami nebude mať Proxmox horší surový výkon VM – v skutočnosti VM v Proxmoxe dosahujú rovnaký výkon ako VM v OpenStacku či CloudStacku na porovnateľnom hardvéri, keďže všetky využívajú KVM/QEMU. Latencia I/O a sieťovej komunikácie VM je minimalizovaná použitím virtio ovládačov a Proxmox podporuje aj techniky ako PCI passthrough alebo SR-IOV pre prípadné zvýšenie výkonu (napr. pre sieťové karty).
- **Výkonnostné limity:** Pri ~30 uzloch v klastri dokáže Proxmox bezpečne prevádzkovať stovky VM (počet VM nie je priamo obmedzený klastrom, skôr hardvérom každého uzla a kapacitou zdieľaného úložiska). V konfigurácii pmxcfs (Proxmox Cluster File System) je uvádzané, že veľkosť databázy configov je limitovaná na 30 MB v RAM, čo postačuje na **konfiguráciu tisícov VM**. Samotný hypervízor KVM dokáže hostiť desiatky VM na jednom uzle (v závislosti od výkonu CPU/RAM), takže z pohľadu **výpočtovej kapacity** Proxmox nie je prekážkou – môžete mať napr. 5 uzlov, každý s 50 VM, spolu 250 VM klaster bez problémov. Úzke hrdlo by však nastalo, ak by ste chceli mať 250 VM *na každý uzol* a k tomu veľa uzlov – vtedy by už management (web GUI, API) mohol reagovať pomalšie, lebo musí agregovať stav stovák VM z mnohých uzlov.
- **Horizontálna škálovateľnosť:** Proxmox VE podporuje klastrovanie, čo umožňuje spravovať viaceré fyzické uzly a distribuovať VM medzi nimi. Klaster Proxmox VE dokáže efektívne spravovať desiatky fyzických uzlov a stovky virtuálnych strojov bez

výrazného poklesu výkonu [39] [42]. Väčšie klastre sú možné, ale vyžadujú precíznejšie plánovanie a správu úložiska.

- **Vertikálna škálovateľnosť:** VM v Proxmox VE môžu byť flexibilne konfigurované s rôznym množstvom CPU, RAM a diskového priestoru, vrátane pridávania zdrojov za behu.
- **Škálovateľnosť:** Navrhnutý pre menšie až stredné prostredia, ako dátové centrá alebo podnikové nastavenia, nie pre veľké cloudové nasadenia. Môže spravovať značný počet VM, ale nie je typicky používaný na stovky alebo tisíce VM naprieč viacerými dátovými centrami. Fórum naznačuje, že chýba natívny nástroj na správu viacerých klastrou, čo obmedzuje škálovateľnosť [41].
- **Využitie zdrojov:** Proxmox VE efektívne využíva KVM hypervisor a LXC kontajnery, čo zabezpečuje dobrý výkon.
- **Latencia:** Latencia pre VM je závislá od podkladového hardvéru a konfigurácie siete. Proxmox priamo neposkytuje cloud-native mechanizmy na optimalizáciu latencie (ako SR-IOV/DPDK abstrakcie), ale tieto technológie sú podporované na úrovni Linuxového jadra, ktoré Proxmox používa. Ich konfigurácia by však bola "mimo" Proxmox GUI.
- **Automatické škálovanie:** Proxmox VE nemá vstavané funkcie automatického škálovania VM na základe zaťaženia. Toto by sa muselo implementovať pomocou externých skriptov alebo nástrojov, ktoré by volali Proxmox API.
- **Reálne použitie:** Populárny medzi menšími organizáciami pre zálohovanie a obnovu po havárii, ale nie bežne používaný pre veľké cloudové služby alebo SOC operácie.

Odôvodnenie: Proxmox VE je schopný spravovať desiatky až stovky VM v klastrovom prostredí s dobrou efektivitou využitia zdrojov. Má dobrú vertikálnu škálovateľnosť VM. Chýbajú mu však natívne mechanizmy automatického škálovania na cloudovej úrovni a pokročilá integrácia pre optimalizáciu latencie na úrovni IaaS. Proxmox VE dosahuje spoľahlivý výkon virtuálnych strojov a je dostatočne škálovateľný pre malé a stredne veľké cloudy, avšak oproti ostatným hodnoteným platformám zaostáva pri veľmi rozsiahlych nasadeniach. Pozitívom je takmer nulová réžia nad rámec KVM (VM bežia s natívnou rýchlosťou) a jednoduchosť klastrovania v menšom rozsahu. Na druhej strane, škálovateľnosť je limitovaná – Proxmox nedokáže lineárne rásť do stovák uzlov kvôli obmedzeniam cluster stacku. Taktiež mu chýbajú natívne mechanizmy automatického škálovania a pokročilého plánovania záťaže, ktoré sú v dynamických cloudových prostrediach žiadané. Zhrnuté, Proxmox je výborne optimalizovaný pre jednoduchosť a výkon na menšom rozsahu, no pre veľké multi-tenant cloudy by bolo potrebné siahnuť po robustnejších riešeniach.

Hodnotenie: 3 b

3.2.1.4 Kubernetes

Popis: Kubernetes je extrémne škálovateľná platforma, hoci primárne pre kontajnery. Jeho výkon a škálovateľnosť sú jednými z jeho najsilnejších stránok:

- **Horizontálna škálovateľnosť:** Kubernetes je navrhnutý pre horizontálne škálovanie podov (kontajnerov) a uzlov (fyzických/virtuálnych serverov) na desiatky tisíc uzlov a státisíce podov [40]. To umožňuje obsluhovať obrovské množstvá súbežných záťaží.
- **Vertikálna škálovateľnosť:** Podporuje vertikálne škálovanie pre kontajnery prostredníctvom konfigurácie limitov CPU/RAM.
- **Využitie zdrojov:** Efektívne využíva zdroje vďaka pokročilému plánovaniu (scheduler) a automatickej obnove.
- **Výkon**: Optimalizovaný pre výkon, s funkciami ako auto-scaling, vyvažovanie záťaže a efektívne riadenie zdrojov. Je vhodný pre reálne záťaže, vrátane monitorovania bezpečnostných udalostí, s blogmi o zlepšeniach výkonu od verzie 1.2

- **Latencia a real-time:** Pre kontajnery je Kubernetes schopný dosiahnuť nízku latenciu a vysokú priepustnosť. Podpora pre SR-IOV a DPDK je možná aj v kontajneroch prostredníctvom špecifických CNI pluginov.
- **Automatické škálovanie:** Má natívne a veľmi robustné mechanizmy automatického škálovania: Horizontal Pod Autoscaler (HPA), Vertical Pod Autoscaler (VPA) a Cluster Autoscaler [41]. Toto je kľúčové pre dynamické prostredia ako SOCaaS.
- **KubeVirt pre VM:** Pri použití KubeVirt, VM bežia v kontajneroch a môžu využívať škálovacie mechanizmy Kubernetes. Avšak, výkon VM v kontajneri môže byť ovplyvnený dodatočnou vrstvou.
- **Výkonnostné testy a limity:** Na overenie škálovateľnosti KubeVirt prebehlo viacero testov. Zaujímavý bol test Red Hatu, kde sa snažili zvýšiť hustotu VM na jeden uzol – dosiahli **400 VM (VMI) na jedinom uzle** so 128 GB RAM. Bolo potrebné zvýšiť predvolené limity (Kubernetes štandardne povoľuje 110 podov na uzol, v teste to zvýšili na ~1022) a vyladiť parametre kubeletu a KubeVirt kontrolérov (napr. zvýšenie limitov API requests z 5 na 50 za sekundu pre KubeVirt kontrolér). Výsledky ukázali, že pri tak extrémnej hustote už narastajú latencie: **100 VM** na uzol sa spustilo za ~3,3 minúty, **200 VM** za ~10 minút a pri 300–400 VM sa čas ustálil okolo 10+ minút. Tieto čísla odrážajú 95. percentil času od požiadavky po plne bežiacu VM. Limitujúcim faktorom sa stal počet vytvorení kontajnerov za minútu – saturácia nastala okolo **48 VM za minútu na uzol**. Zároveň sa tým potvrdilo, že škálovať do 500 VM na uzol už nešlo kvôli vyčerpaniu zdrojov (kontajner runtime začal zlyhávať pri pokuse spustiť viac). Tento test je však extrém skôr na overenie kontrolnej roviny; v praxi by sa VM rozložili medzi mnoho uzlov, takže reálne **KubeVirt zvládne stovky až tisíce VM paralelne** pokiaľ sú distribuované. Kubernetes etcd databáza unesie spomínaných ~150k objektov, čiže aj keby každá VM mala 2–3 objekty (VM, Pod, Service), klaster s 5000 uzlami by vedel spravovať rádovo desaťtisíce VM.
- **Reálne použitie:** Široko používaný veľkými poskytovateľmi cloudov a podnikmi pre správu kontajnerizovaných aplikácií, vrátane SOC operácií, kde je kritické reálne monitorovanie a reakcia. Google Kubernetes Engine (GKE), Amazon EKS a Azure AKS sú príklady masívne škálovateľných produkčných nasadení Kubernetes.

Odôvodnenie: Kubernetes exceluje v škálovateľnosti, predovšetkým pre kontajnerizované záťaže. Ponúka natívne a robustné mechanizmy horizontálneho aj vertikálneho automatického škálovania a je navrhnutý pre obrovské rozsahy. Hoci jeho primárnym zameraním nie sú VM, jeho základné vlastnosti a schopnosť škálovať workloady (vrátane VM cez KubeVirt) ho robia extrémne vhodným pre dynamické a rastúce požiadavky SOCaaS.

Hodnotenie: 5 b

3.2.2 K5: Spoľahlivosť a stabilita (váha: 4)

Odôvodnenie: Posudzujeme mieru vyzretosti platformy, jej stabilitu v produkčnej prevádzke a dostupnosť mechanizmov pre vysoko dostupné nasadenie. Kritérium zahŕňa úroveň otestovania a odladenia softvéru a možnosť nasadiť komponenty redundantne (HA – High Availability). V prostredí bezpečnostného centra je dôležité minimalizovať prestoje a zaistiť nepretržitú dostupnosť služby. Pri hodnotení cloudových služieb sa odporúča dôkladne preskúmať minulú výkonnosť, spokojnosť zákazníkov a zvládanie neplánovaných výpadkov. Rovnako porovnávame SLA (Service Level Agreement) ak sú k dispozícii – hoci v prípade open-source platformy ide skôr o interné garancie, dostupnosť podpory atď.

Keďže spoľahlivosť priamo ovplyvňuje kvalitu poskytovanej SOC služby, má toto kritérium **váhu 4** (vyššia dôležitosť).

Odporúčanie pre hodnotenie:

- 5 bodov: Bez vážnych výpadkov, spoľahlivé referencie v produkcii, podpora HA nasadenia.
- 3 body: Občasné menšie výpadky, čiastočná podpora HA.
- 1 bod: Časté alebo významné výpadky, žiadna podpora HA.

3.2.2.1 OpenStack

Popis: OpenStack je dlho vyvíjaná platforma, ktorá sa vyvíja už viac ako 14 rokov a je rozsiahle nasadená v kritických produkčných prostrediach po celom svete [46]. Spoľahlivosť a vysoká dostupnosť sú základnými piliermi jeho dizajnu. Dnes patrí k najrozšírenejším open-source cloudovým platformám: podľa prieskumov beží cez 40 miliónov jadier VM v produkčných OpenStack cloudoch po celom svete [47]. Referencie potvrdzujú spoľahlivosť – OpenStack používa napr. AT&T pre mobilnú sieť core, Walmart prevádzkuje takmer milión jadier, China Mobile buduje NFV sieť s 50 tisíc servermi. Taktiež firmy ako Blizzard (herná infraštruktúra) či PayPal (vyše miliarda transakcií denne) využívajú OS. Veľké vedecko-akademické clustre (CERN, NSF JetStream) tiež potvrdili stabilitu a spoľahlivosť OpenStacku vo veľkom meradle:

- **Vyzretosť a otestovanie:** Vďaka obrovskej globálnej komunite a rozsiahlym produkčným nasadeniam (vrátane dátových centier veľkých podnikov a telekomunikačných operátorov) je OpenStack dôkladne testovaný a odladený. Každá nová verzia prechádza prísnymi testovacími cyklami [46].
- **Vysoká dostupnosť (HA):** OpenStack podporuje nasadenie v HA móde na viacerých úrovniach. Riadiace služby (kontrolné uzly) je možné prevádzkovať redundantne (typicky 3+ kontrolery) s použitím klastrovacích technológií ako Pacemaker a load-balancera HAProxy, čím sa eliminuje single point of failure. Riadiaca rovina tak môže bežať v režime *active/active* (viac inštancií služby naraz) alebo *active/passive* podľa povahy komponentu . Databázová vrstva využíva replikačný klaster (napr. Galera pre MariaDB) pre odolnosť voči výpadku DB uzla. Výpočtové uzly: samotné výpadky hypervízorov neovplyvnia chod ostatných VM; OpenStack má mechanizmy na automatickú obnovu VM. Komponent Masakari poskytuje *high availability* pre inštancie tým, že pri zlyhaní hypervízora automaticky reštartuje dotknuté VM na živých uzloch.
- **Podpora HA nasadenia:** Všetky kľúčové komponenty OpenStacku (Nova API, Neutron API, Cinder API, Keystone, Glance, atď.) môžu byť nasadené redundantne v konfigurácii s vysokou dostupnosťou (HA). To sa typicky dosahuje pomocou:
    - Viacerých inštancií riadiacich služieb za load balancerom.
    - Klastrových riešení pre databázy (napr. Galera Cluster pre MySQL) a správu správ (napr. RabbitMQ cluster).
    - Distribuovaných súborových systémov a úložných riešení (Ceph, Swift) pre odolnosť voči zlyhaniu úložiska.
    - Mechanizmov prevencie "single point of failure" na úrovni sietí a výpočtov.

Odôvodnenie: OpenStack je vysoko vyzretá platforma s preukázanou stabilitou v rozsiahlych produkčných nasadeniach. Ponúka komplexné a robustné mechanizmy pre implementáciu vysokej dostupnosti pre všetky svoje komponenty, minimalizujúc prestoje a zabezpečujúc nepretržitú dostupnosť služby SOCaaS.

Hodnotenie: 5 b

3.2.2.2 CloudStack

Popis: CloudStack je tiež zavedená platforma, ktorá sa etablovala ako spoľahlivé riešenie pre IaaS, najmä v prostredí poskytovateľov služieb:

- **Vyzretosť a otestovanie:** CloudStack je open-source projekt pod záštitou Apache Software Foundation s dlhou históriou vývoja a rozsiahlym používaním v produkcii. To zabezpečuje vysokú úroveň otestovania a odladenia [48].
- **Podpora HA nasadenia:** Manažment server CloudStacku (ktorý je potenciálnym single point of failure) môže byť nasadený v režime vysokej dostupnosti (HA) s aktívno-pasívnou alebo aktívno-aktívnou konfiguráciou [49]. Podporuje HA aj pre databázy a sieťové služby. Samotné VM na hypervizoroch sú chránené mechanizmami vysokou dostupnosťou hypervizora (napr. VMware HA, alebo KVM s Pacemaker/Corosync).
- **Referencie v produkcii:** Mnoho poskytovateľov cloudu a veľkých podnikov používa CloudStack pre kritické služby, čo svedčí o jeho stabilite a spoľahlivosti v produkčnom prostredí (napr. T-Mobile, NTT Communications, Citrix). Aj keď nie tak rozsiahle medializované ako OpenStack, referencie sú solídne [50]

Odôvodnenie: CloudStack je vyzretá a spoľahlivá platforma s preukázanou stabilitou v produkčných prostrediach. Ponúka robustné mechanizmy pre vysokú dostupnosť manažment servera a podporuje HA funkcie pre hostiteľov, čo zaisťuje nepretržitú dostupnosť služieb SOCaaS.

Hodnotenie: 5 b

3.2.2.3 Proxmox VE

Popis: Proxmox VE je stabilná a robustná virtualizačná platforma, ktorá poskytuje vysokú dostupnosť na úrovni klastra pre virtuálne stroje:

- **Vyzretosť a otestovanie:** Proxmox VE má aktívnu komunitu a stabilný vývojový cyklus s pravidelnými aktualizáciami, čo naznačuje dobrú úroveň otestovania softvéru [51]. Je široko používaný v menších a stredných firmách, ako aj v akademickom sektore.
- **Podpora HA nasadenia:** Proxmox VE má vstavaný klaster s vysokou dostupnosťou (HA) založený na Corosync a Pacemaker, ktorý umožňuje automatickú migráciu virtuálnych strojov a kontajnerov na iný funkčný uzol v prípade zlyhania hardvéru [52]. To zabezpečuje vysokú dostupnosť samotných VM. Manažment vrstva (webové rozhranie a API) je distribuovaná v rámci klastra a pri zlyhaní jedného uzla je dostupná cez iný.
- **Referencie v produkcii:** Proxmox je spoľahlivý pre nasadenia v rozsahu desiatok až stoviek VM. Hoci možno nemá referencie od gigantických firiem ako OpenStack, jeho stabilita v reálnych produkčných prostrediach je všeobecne uznávaná v rámci jeho cieľového segmentu.

Odôvodnenie: Proxmox VE je stabilná platforma s robustnými vstavanými mechanizmami pre vysokú dostupnosť VM v klastri. Hoci je skôr virtualizačným riešením než plnohodnotnou cloudovou platformou, jeho schopnosť zabezpečiť nepretržitú prevádzku virtuálnych komponentov pre SOCaaS je veľmi dobrá a preukázaná v mnohých produkčných prostrediach.

Hodnotenie: 4,5 b

3.2.2.4 Kubernernetes

Popis: Kubernetes je od základu vyvíjaný pre vysokú dostupnosť, odolnosť voči zlyhaniu a samoopravné schopnosti, čo je kritické pre mikroslužbové architektúry a dynamické prostredia:

- **Vyzretosť a otestovanie:** Kubernetes je de facto štandardom pre orchestráciu kontajnerov, používaný tisíckami firiem po celom svete, vrátane tých najväčších. Je intenzívne testovaný a neustále vylepšovaný obrovskou komunitou a firmami ako Google, Red Hat atď. [53].
- **Podpora HA nasadenia:** Celá riadiaca rovina (Control Plane) Kubernetes môže byť nasadená vo vysoko dostupnej konfigurácii [10], kde komponenty ako API Server, etcd, Scheduler a Controller Manager bežia redundantne. Pracovné uzly (Worker Nodes) sú tiež redundantné a Kubernetes automaticky preplánuje pody zlyhaných uzlov na zdravé. Vstavané mechanizmy ako "self-healing" (reštart padnutých kontajnerov) a "liveness/readiness probes" zabezpečujú nepretržitú dostupnosť aplikácií.
- **Referencie v produkcii:** Milióny aplikácií bežia na Kubernetes v kritických produkčných prostrediach s extrémne vysokými požiadavkami na SLA (napr. Spotify, Airbnb, IBM, Google Cloud, AWS, Azure) [53]. Tieto referencie potvrdzujú jeho vynikajúcu spoľahlivosť a stabilitu.
- **KubeVirt stabilita:** Pri použití KubeVirt pre VM, je stabilita VM ovplyvnená stabilitou Kubernetes klastra a samotného KubeVirt. Keďže KubeVirt využíva natívne mechanizmy Kubernetes, profituje z jeho HA schopností.

Odôvodnenie: Kubernetes je od základu navrhnutý pre extrémne vysokú spoľahlivosť a stabilitu. Jeho architektonické princípy zahŕňajú redundanciu riadiacej roviny, samoopravné schopnosti a robustné mechanizmy pre automatickú obnovu workloadov. Je de facto štandardom pre kritické produkčné nasadenia, čo zabezpečuje nepretržitú dostupnosť komponentov SOCaaS.

Hodnotenie: 5 b

3.2.3 K6: Súlad so štandardmi a interoperabilita (váha: 4)

Odôvodnenie: Z technického hľadiska je významné, či platforma podporuje otvorené štandardy a bežné technológie, čo podmieňuje jej interoperabilitu s inými systémami. Riešenie by malo byť kompatibilné s populárnymi hypervízormi, sieťovými protokolmi, úložiskovými technológiami a rozhraniami pre správu (napr. OpenStack API je de facto štandard pre open-source cloudy). Široká podpora štandardov zároveň uľahčuje integráciu a minimalizuje riziko proprietárnych obmedzení. Posudzujeme tiež dostupnosť certifikácií alebo referencií – napríklad či platforma vychádza z osvedčených open-source projektov podporovaných veľkou komunitou. Ako naznačujú postupy hodnotenia cloudových poskytovateľov, dodržiavanie priemyselných štandardov a osvedčení (napr. ISO 27001 v bezpečnosti) je významným kritériom výberu. V našom projekte interoperabilita umožní v budúcnosti prepojenie viacerých cloudových prostredí a integráciu ďalších komponentov.

Je tiež dôležité, aby platforma obsahovala nástroje na implementáciu požadovaných opatrení (napr. logovanie, šifrovanie, prístupové kontroly) a aby komunita/dodávatelia poskytovali usmernenia pre súlad s predpismi. Súlad s reguláciami ovplyvní prijateľnosť riešenia v produkcii – napríklad pre dáta verejnej správy by platforma mala podporovať nasadenie v súlade so zákonom o ochrane osobných údajov (GDPR). Vzhľadom na charakter plánovaného SOC riešenia (ktoré môže spracúvať citlivé bezpečnostné údaje klientov) považujeme toto kritérium za veľmi významné a priraďujeme mu váhu 4.

Odporúčanie pre hodnotenie:

- **5 bodov:** Plná kompatibilita s otvorenými štandardmi, certifikácie (ISO 27001, GDPR compliant), aktívna interoperabilita.
- **3 body:** Čiastočná kompatibilita a interoperabilita, niektoré štandardy nie sú úplne pokryté.
- **1 bod:** Obmedzená alebo žiadna kompatibilita, problémy s interoperabilitou.

3.2.3.1 OpenStack

Popis: OpenStack je open-source IaaS platforma s veľmi dobrou podporou štandardov a interoperability. OpenStack je navrhnutý ako otvorená cloudová platforma s dôrazom na dodržiavanie priemyselných štandardov. Podporuje viacero hypervízorov (KVM, Xen, VMware, Hyper-V) a využíva osvedčené open-source technológie ako libvirt či Open vSwitch. Jeho API rozhrania sa stali de-facto štandardom pre open-source cloudy – napríklad mnoho nástrojov tretích strán podporuje OpenStack API a OpenStack samotný využíva známe formáty (napr. kompatibilitu so S3 pre objektové úložisko alebo s CloudFormation šablónami v orkchestrácii) [56]. Vďaka širokej podpore štandardných protokolov (HTTP/REST API, POSIX pre image, atď.) a otvorenému vývoju je integrácia OpenStacku s existujúcimi systémami a službami plynulá. Poskytuje:

- **Súlad so štandardmi**: OpenStack má rozsiahlu dokumentáciu o súlade so štandardmi, vrátane ISO 27001, SOC 1/2 a GDPR (OpenStack Compliance Overview). Jeho Keystone identity service podporuje silnú autentifikáciu a prístupové politiky, ktoré pomáhajú splniť regulačné požiadavky, ako je PCI-DSS (Red Hat OpenStack PCI-DSS).
- **Interoperabilita**: Podporuje širokú škálu hypervizorov (napr. KVM, Xen, VMware), úložiskových systémov (napr. Ceph, Cinder) a sieťových technológií (napr. Neutron). OpenStack API je považované za de facto štandard v open-source cloudoch, čo uľahčuje integráciu s inými systémami (OpenStack Security Guide).
- **Certifikácie a komunita**: OpenStack má veľkú komunitu a pravidelné aktualizácie, ktoré podporujú súlad s normami a poskytujú usmernenia pre bezpečnosť a regulácie (OpenStack Cloud Security).

Odôvodnenie: Plne spĺňa požiadavky na súlad s otvorenými štandardmi a interoperabilitu. OpenStack API je široko prijatý štandard. Disponuje robustným ekosystémom nástrojov a podpory, ktoré sú nevyhnutné pre dosiahnutie a preukázanie súladu s medzinárodnými certifikáciami (ISO 27001) a regulačnými požiadavkami (GDPR) pre multitenantné prostredie spracúvajúce citlivé bezpečnostné dáta. OpenStack má jednu z najväčších open-source komunít a je považovaný za globálny štandard pre IaaS cloudy – nasadený je vo veľkých podnikoch aj v telekomunikačnom sektore. Rešpektuje princípy interoperability: opakovane používa existujúce rozhrania a formáty (napr. AWS S3 API v rámci Swiftu) a jeho vlastné komponentové API sú široko podporované v nástrojoch a knižniciach. Nezávislé testy interoperabilty (Interop program OpenInfra Foundation) zaručujú, že rôzne distribúcie OpenStacku spĺňajú jednotné požiadavky, čo uľahčuje prenos aplikácií medzi prostrediami. OpenStack tak minimalizuje riziko viazanosti na dodávateľa a maximalizuje súlad s bežnými štandardmi cloudového odvetvia

Hodnotenie: 5 b

3.2.3.2 CloudStack

Popis: CloudStack tiež podporuje rozsiahlu sadu virtualizačných technológií (KVM, VMware ESXi, XenServer) a poskytuje konzistentné API pre správu infraštruktúry. Jeho API je tiež RESTful a dobre zdokumentované, čo umožňuje integráciu s inými systémami. Komunita CloudStacku je menšia v porovnaní s OpenStackom, ale stále je aktívna a existuje komerčná podpora. Z hľadiska jednoduchosti nasadenia a prevádzky pre menšie a stredné prostredia je často preferovaný. Čo sa týka súladu s reguláciami, CloudStack ponúka základné funkcie ako RBAC a sieťovú izoláciu, no hĺbka a rozsah nástrojov, usmernení a certifikácií špecifických pre súlad s prísnymi požiadavkami (ako detailné audity pre ISO 27001 alebo špecifické požiadavky GDPR pre verejnú správu) môžu byť menej rozvinuté ako u OpenStacku. Dosiahnutie plného súladu by mohlo vyžadovať viac manuálnej práce a externých nástrojov. Poskytuje:

- **Súlad so štandardmi**: Existuje certifikácia ACCEL (Apache CloudStack Certification Exam by LPI-JAPAN), ktorá overuje znalosti a schopnosti pri práci s CloudStack (ACCEL Certification). Informácie o špecifických štandardoch, ako ISO 27001 alebo GDPR, sú však obmedzené, čo naznačuje čiastočnú kompatibilitu.
- **Interoperabilita**: Podporuje rôzne hypervizory (napr. VMware, KVM, XenServer) a úložiskové systémy (napr. NFS, Ceph), ale jeho ekosystém nie je tak rozsiahly ako u OpenStack (Apache CloudStack).
- **Certifikácie a komunita**: CloudStack má menšiu komunitu ako OpenStack, ale je aktívne udržiavaný Apache Software Foundation, čo zaisťuje určitú úroveň podpory a aktualizácií.

Odôvodnenie: CloudStack kladie dôraz na jednoduchosť integrácie – v základnej inštalácii poskytuje jednotný balík služieb, ktorý využíva osvedčené open-source komponenty. Vďaka voliteľnej kompatibilite s AWS API zvyšuje interoperabilitu so systémami pôvodne určenými pre Amazon cloud [57]. Komunita Apache zabezpečuje transparentný vývoj a absenciu vendor lock-in: všetci používatelia používajú ten istý open-source kód, čo zvyšuje vzájomnú kompatibilitu nasadení. Hoci komunita CloudStack je menšia než pri OpenStacku, platforma dodržiava hlavné štandardy v oblasti virtualizácie a sieťovania. Menším nedostatkom môže byť užší ekosystém doplnkov, no v rámci štandardnej funkcionality (hypervízory, siete, úložiská) CloudStack nezaostáva.

Hodnotenie: 3,5 b

3.2.3.3 Proxmox VE

Popis: Proxmox podporuje štandardné virtualizačné protokoly, využíva hypervízor KVM (integrálny súčasť Linux jadra) a kontajnery LXC, ktoré patria k priemyselným štandardom v virtualizácii. Podporuje bežné formáty diskových obrazov (QCOW2, RAW) a štandardné úložiskové technológie (LVM, ZFS, Ceph). Prostredníctvom REST API (s JSON formátom) umožňuje jednoduchú integráciu s nástrojmi tretích strán

Neponúka však rozsiahle cloudové API kompatibilné s OpenStack API alebo inými cloudovými štandardmi pre dynamickú, programovateľnú orchestráciu cloudových služieb. Jeho webové rozhranie a CLI sú primárnymi nástrojmi pre správu. Interoperabilita s komplexnými cloudovými prostrediami by si vyžadovala značné úsilie a budovanie nadstavbových integračných vrstiev. Čo sa týka súladu s reguláciami (ISO 27001, GDPR), Proxmox VE neposkytuje natívne nástroje alebo certifikačné programy na úrovni celej cloudovej platformy. Zabezpečenie súladu by bolo v značnej miere zodpovednosťou implementátora a vyžadovalo by kombináciu vlastných procesov, externých nástrojov a starostlivej konfigurácie operačného systému,

Poskytuje:

- **Súlad so štandardmi**: Neexistujú špecifické certifikácie ako ISO 27001 alebo GDPR, ale Proxmox môže byť tvrdený pre bezpečnosť a súlad s normami, ako je uvedené na fóre (Proxmox Hardening). Podpora CIS Benchmarkov naznačuje potenciál pre súlad (Proxmox CIS Benchmark).
- **Interoperabilita**: Podporuje rôzne úložiskové technológie (napr. Ceph, ZFS, NFS) a sieťové protokoly, ale nie je navrhnutý pre rozsiahle cloudové nasadenia, čo obmedzuje jeho interoperabilitu (Proxmox VE Administration).
- **Certifikácie a komunita**: Proxmox má aktívnu komunitu, ale menšiu ako OpenStack alebo Kubernetes, s obmedzenými usmerneniami pre regulácie.

Odôvodnenie: Má obmedzenú alebo žiadnu kompatibilitu s otvorenými cloudovými štandardami IaaS pre multitenantné prostredie. Je to primárne virtualizačná platforma založené čisto na open-source komponentoch (Debian, KVM, QEMU, LXC).jProxmox vysoko kompatibilný s bežnými štandardmi virtualizácie, nie komplexné cloudové riešenie. Chýbajú mu štandardizované cloudové API pre rozsiahlu orchestráciu a natívna podpora pre komplexné regulačné požiadavky (ako sú ISO 27001 a GDPR na úrovni cloudovej platformy), čo by viedlo k problémom s interoperabilitou a súladom v projekte SOC.

Hodnotenie: 3 b

3.2.3.4 Kubernernetes

Popis: Kubernetes je popredný otvorený štandard pre orchestráciu kontajnerov a cloud-native aplikácií. Je postavený na modulárnej architektúre s bohatým a flexibilným API, ktoré je *de facto* štandardom pre kontajnerové platformy. Podporuje širokú škálu otvorených štandardov (OCI pre kontajnery, CNI pre sieť, CSI pre úložisko) a má mimoriadne aktívnu a rozsiahlu globálnu komunitu. V rámci ekosystému Kubernetes existuje množstvo nástrojov a operátorov, ktoré uľahčujú implementáciu bezpečnostných opatrení (segmentácia siete, prístupové kontroly, šifrovanie) a pomáhajú pri dosahovaní súladu s reguláciami (ISO 27001, GDPR) pre kontajnerizované aplikácie. Mnoho komerčných Kubernetes distribúcií a spravovaných služieb je certifikovaných. Avšak, hoci je možné spravovať virtuálne stroje v Kubernetes prostredníctvom projektov ako KubeVirt, nie je to jeho primárna funkcia a pridáva to vrstvu komplexnosti. Pre SOC riešenie postavené primárne na IaaS (virtuálne stroje) by to nebolo najpriamejšie riešenie z hľadiska natívnych štandardov. Pre kontajnerizované SOC komponenty by však bol ideálny.

Poskytuje:

- **Súlad so štandardmi**: Podporuje rámce ako CIS Benchmarks, NIST SP 800-53 a 800-190, ktoré uľahčujú súlad s reguláciami ako GDPR (Kubernetes Compliance Frameworks). Pod Security Standards zaisťujú bezpečnostné politiky (Kubernetes Pod Security).
- **Interoperabilita**: Podporuje rôzne kontejnerové runtimes (napr. containerd, CRI-O), úložiskové systémy (cez CSI) a sieťové riešenia (cez CNI), čo zaisťuje vysokú interoperabilitu (Kubernetes Security).
- **Certifikácie a komunita**: Kubernetes má veľkú komunitu pod záštitou CNCF, s pravidelnými aktualizáciami a usmerneniami pre súlad s reguláciami (Kubernetes NIST Guide).

Odôvodnenie: Kubernetes ako otvorený projekt má obrovskú komunitu a takmer univerzálnu podporu – podľa prieskumov CNCF ho v produkcii využíva okolo 80% organizácií [58]. To zaručuje širokú interoperabilitu: aplikácie kontajnerizované pre Kubernetes je možné

jednoducho presúvať medzi rôznymi prostrediami. KubeVirt stavia na tých istých princípoch, čím prináša VM do sveta Kubernetes bez uzavretých formátov. Hoci samotný Kubernetes nemá vstavaný komponent na správu VM, KubeVirt je projektom CNCF (vydanie v1.0 v roku 2023) a riadi sa osvedčenými postupmi komunitného vývoja. Integrácia s existujúcimi nástrojmi (CI/CD, Istio service mesh, atď.) prebieha cez otvorené API Kubernetes. Avšak, pre základné IaaS funkcionality (správa virtuálnych strojov), ktoré sú pre tento projekt definované ako kľúčové pre multitenantné SOC, je Kubernetes skôr rozšírením (cez KubeVirt) než natívnym riešením. To spôsobuje, že niektoré IaaS štandardy nie sú priamo pokryté, čo vedie k "čiastočnej" interoperabilite v kontexte potrieb projektu. Miernym limitom môže byť komplexnosť riešenia pri snahe dosiahnuť plnú kompatibilitu so všetkými IaaS funkciami, no z hľadiska štandardov a interoperability Kubernetes + KubeVirt výrazne boduje ako moderná, otvorená platforma

Hodnotenie: 4 b

3.3 PREVÁDZKOVÉ KRITÉRIÁ

3.3.1 K7: Náročnosť nasadenia a správy (váha: 3)

Odôvodnenie: Toto kritérium posudzuje, aké úsilie a zdroje bude potrebné vynaložiť na inicializáciu platformy a jej následnú správu. Vstupuje sem zložitosť inštalácie (počet a komplikovanosť krokov pri nasadzovaní celého prostredia), požiadavky na odborné znalosti personálu, čas potrebný na plné sprevádzkovanie, ale aj prehľadnosť údržby (aktualizácie, upgrady, škálovanie). Napríklad niektoré open-source platformy sú vysoko konfigurovateľné, ale môžu byť komplexné na nasadenie, čo si vyžaduje skúsený tím alebo využitie automatizačných nástrojov (napr. Ansible skripty a pod.). Z prevádzkového hľadiska hodnotíme aj to, či riešenie prináša funkcionality uľahčujúce správu – napr. self-healing, monitoring, jednoduché rozšírenie klasteru apod. Cieľom je zvoliť platformu, ktorej prevádzka nebude pre organizáciu neúmernou záťažou.

Tomuto kritériu dávame **váhu 3** (stredná dôležitosť), vzhľadom na to, že niektoré dodatočné úsilie pri open-source riešení je akceptovateľné, pokiaľ ostatné prínosy prevažujú.

Odporúčanie pre hodnotenie:

- 5 bodov: Jednoduché nasadenie, jasná dokumentácia, minimálne požiadavky na správu, silná automatizácia.
- 3 body: Mierne komplexnejšie nasadenie, zvládnuteľné s dostupnými nástrojmi a dokumentáciou.
- 1 bod: Zložitá inštalácia, náročná administrácia, nedostatočná dokumentácia.

3.3.1.1 OpenStack

Popis: Nasadenie OpenStacku je známe svojou komplexnosťou, najmä pre produkčné prostredia. Vyžaduje si hlboké znalosti rôznych komponentov (Nova, Neutron, Cinder, Glance, Keystone atď.), konfigurácie siete, úložiska a integrácie s hypervízormi. Existuje mnoho inštalačných metód (TripleO, Kolla-Ansible, OpenStack-Ansible, Packstack), ktoré síce automatizujú proces, ale stále vyžadujú značné odborné znalosti a skúsenosti na správne plánovanie, nasadenie a ladenie. Údržba, aktualizácie a upgrady sú tiež náročné operácie, ktoré si vyžadujú starostlivé plánovanie a môžu byť časovo náročné. OpenStack však ponúka rozsiahle API pre automatizáciu a existujú nástroje pre monitoring a čiastočný self-healing. Pre plnohodnotnú správu a škálovanie je potrebný skúsený tím DevOps/cloud inžinierov. V komerčných distribúciách je správa čiastočne uľahčená, ale základná komplexnosť zostáva.

- **Nasadenie**: Zložité kvôli viacerým komponentom (Nova, Neutron, Cinder), vyžaduje odborné znalosti a automatizačné nástroje ako Ansible alebo Terraform. Inštalácia môže trvať dni až týždne (OpenStack Deployment Guide).
- **Správa**: Náročná kvôli potrebe manuálnej konfigurácie a údržby, hoci nástroje ako Heat a monitoringové riešenia (napr. Zabbix) uľahčujú správu. Dokumentácia je rozsiahla, ale vyžaduje technické zručnosti (OpenStack Operations Guide).
- **Automatizácia**: Silná podpora automatizácie cez Heat, Ansible a Terraform, s funkciami ako self-healing pri integrácii s externými nástrojmi.

Odôvodnenie: Zložitá inštalácia a náročná administrácia pre plnohodnotné produkčné prostredie. Vyžaduje si vysokú úroveň odborných znalostí a značné zdroje na nasadenie, údržbu a riadenie životného cyklu, napriek existujúcim automatizačným nástrojom. V porovnaní s ostatnými platformami je nasadenie OpenStacku považované za komplexné – jeho flexibilita prichádza s cenou vyššej administratívnej záťaže. túdie uvádzajú, že na spoľahlivé prevádzkovanie je potrebný väčší tím s hlbokou znalosťou komponentov. Na druhej strane, pre veľké nasadenia je táto investícia odôvodnená robustnosťou výsledného cloudu.

Hodnotenie: 1,5 b

3.3.1.2 CloudStack

Popis: CloudStack poskytuje **“all-in-one”** balík – po inštalácii manažérskeho servera (Java aplikácia + databáza) a agentov na hypervízoroch je základné prostredie pripravené. Predkonfigurovaný stack služieb znižuje počet krokov potrebných na sprevádzkovanie cloudu. Oproti OpenStacku má CloudStack menej samostatných komponentov, čo zjednodušuje architektúru (napr. sieťové a úložiskové funkcie sú integrované priamo). Inštalácia typicky prebieha cez balíčky (RPM/DEB) podľa príručky, pričom pre menšie nasadenia môže aj jediný manažér obslúžiť cloud. Správa systému (užívatelia, zóny, šablóny) sa dá pohodlne robiť cez webové rozhranie.

Odôvodnenie: V porovnaní s OpenStackom je nasadenie CloudStacku výrazne menej komplexné – literatúra uvádza, že CloudStack je **„simple to set up“** a vhodný aj pre menšie tímy [59] [60] . Integrovaný prístup (vopred pripravené kombinácie služieb) odbúrava nutnosť zdĺhavých konfiguračných zásahov a ladenia kompatibility medzi komponentmi. Organizácie bez rozsiahlych skúseností s cloudom vedia CloudStack rozchodiť rýchlejšie, čo skracuje čas na uvedenie do prevádzkyopenmetal.io. Menšia modularita síce obmedzuje možné prispôsobenia, ale zároveň znamená menej možností, kde spraviť chybné nastavenie. Celkovo je správa a aktualizácia CloudStacku menej náročná, upgradovací cyklus je pomalší (LTS verzie) a dokumentácia pokrýva najbežnejšie scenáre nasadenia.

Hodnotenie: 4 b

3.3.1.3 Proxmox VE

Popis: Proxmox VE je známy pre svoju jednoduchú inštaláciu. Typická inštalácia z ISO obrazu je priamočiara a rýchla. Má intuitívne webové rozhranie, ktoré značne uľahčuje správu virtuálnych strojov, kontajnerov, úložiska a siete. Pridávanie ďalších uzlov do klastra a škálovanie je pomerne jednoduché. Pre údržbu a aktualizácie existuje jasná dokumentácia a procesy. Nevyžaduje rozsiahle špecializované IT znalosti nad rámec základnej virtualizácie a sieťovania. Funkcie ako live migration, integrovaný backup a HA (High Availability) klaster uľahčujú prevádzku. Automatizácia je možná prostredníctvom CLI a API, ale nie je tak rozsiahla pre cloudové služby ako v prípade OpenStacku. Je to však veľmi user-friendly riešenie pre virtualizačnú vrstvu.

- **Nasadenie**: Veľmi jednoduché, s inštaláciou cez ISO obraz, dokončiteľnou za hodiny. Webové rozhranie uľahčuje konfiguráciu (Proxmox Installation).
- **Správa**: Intuitívne webové rozhranie nevyžaduje hlboké odborné znalosti. Podpora aktualizácií a škálovania je jednoduchá, s integrovaným monitoringom (Proxmox VE Administration Guide).
- **Automatizácia**: Podpora automatizácie cez REST API a Ansible, hoci menej pokročilá ako OpenStack alebo Kubernetes.

Odôvodnenie: Jednoduché nasadenie z ISO, intuitívne webové rozhranie a jasná dokumentácia pre správu. Minimálne požiadavky na správu a jednoduché škálovanie pre virtualizačné prostredie. Silná automatizácia pre základné úlohy virtualizácie.

Hodnotenie: 5 b

3.3.1.4 Kubernetes

Popis: Nasadenie Kubernetes klastra, najmä v produkčnom prostredí, môže byť extrémne komplexné. Existuje mnoho spôsobov inštalácie (kubeadm, kops, Rancher, OpenShift, cloudové služby ako GKE/EKS/AKS), pričom každý má svoje špecifiká a vyžaduje si hlboké znalosti Linuxu, kontajnerov, siete (CNI), úložiska (CSI) a orchestrácie. Správa a údržba Kubernetes klastra si vyžaduje špecializovaných inžinierov s rozsiahlymi zručnosťami v oblasti cloud-native technológií. Upgrady môžu byť náročné. Na druhej strane, Kubernetes je navrhnutý pre vysokú úroveň automatizácie (deklaratívne API, operátory), self-healing (reštartovanie zlyhaných podov, automatické škálovanie) a monitoring. Akonáhle je klaster nasadený a spustený, správa aplikácií v ňom môže byť efektívna, ale počiatočná investícia do nasadenia a prevádzky základnej infraštruktúry je značná.

- **Nasadenie**: Zložité, najmä pre HA klastre, vyžaduje odborné znalosti a nástroje ako kubeadm alebo KubeSpray. Inštalácia môže trvať dni (Kubernetes Setup).
- **Správa**: Náročná kvôli komplexnosti konfigurácie, ale nástroje ako operátory, Helm a monitoring (napr. Prometheus) uľahčujú správu. Dokumentácia je rozsiahla (Kubernetes Documentation).
- **Automatizácia**: Silná podpora self-healing, automatického škálovania a monitoringu, čo znižuje prevádzkovú záťaž po nasadení.

Odôvodnenie: Zložitá inštalácia a náročná administrácia pre rozsiahle produkčné prostredie. Vyžaduje si špecializované odborné znalosti a značné zdroje na nasadenie, údržbu a riadenie životného cyklu klastra, napriek vynikajúcim natívnym funkciám pre automatizáciu a self-healing.

Hodnotenie: 1 b

3.3.2 K8: Podpora, dostupnosť dokumentácie a komunita (váha: 4)

Odôvodnenie: Úroveň dostupnej podpory pre danú platformu výrazne ovplyvňuje jej praktickú použiteľnosť. Pri komerčných riešeniach sledujeme kvalitu technickej podpory dodávateľa (dostupnosť 24/7, SLA pre riešenie incidentov). Pri open-source riešeniach je dôležitá veľkosť a aktivita komunity – počet prispievateľov, frekvencia vydávania nových verzií a opráv chýb, dostupnosť dokumentácie a znalostnej bázy. Silná komunita a podpora od veľkých IT firiem sú zárukou dlhodobej udržateľnosti projektu. Hodnotí sa aj existencia školení, certifikácií alebo konzultačných služieb pre danú platformu. Podľa skúseností môžu menší poskytovatelia ponúkať osobnejšiu podporu, kým väčší hráči majú rozsiahlejšie zdroje podpory, avšak často za príplatok. Keďže v našom prípade uvažujeme open-source platformu,

kritérium sa sústreďuje na komunitnú podporu a ekosystém okolo nej. Hodnotí sa vývojová aktivita projektu, frekvencia aktualizácií

Priraďujeme mu **váhu 4**, keďže dostupnosť odbornej podpory a aktívny vývoj sú pre úspech projektu veľmi dôležité.

Odporúčanie pre hodnotenie:

- **5 bodov:** Aktívna komunita, bohatá dokumentácia, pravidelné aktualizácie a profesionálna podpora.
- **3 body:** Priemerná komunita, základná dokumentácia, pomalšie aktualizácie.
- **1 bod:** Slabá alebo neexistujúca komunita, zastaraná dokumentácia, minimálna podpora.

3.3.2.1 OpenStack

Popis: OpenStack má mimoriadne širokú a aktívnu komunitu – do projektu prispeli tisíce vývojárov po celom svete a pod hlavičkou OpenInfra Foundation je registrovaných vyše 110 000 členov v 187 krajinách. Projekt má výbornú dokumentáciu: oficiálne príručky pokrývajú inštaláciu, architektúru aj vývoj API (napr. detailný **OpenStack Documentation** portál). Dostupné sú aj knihy, online kurzy a certifikácie (COA – Certified OpenStack Administrator). Veľké firmy (Red Hat, Canonical, Mirantis) ponúkajú platenú podporu a pravidelné vydania (distribúcie OpenStacku s dlhodobou podporou). Komunita organizuje každoročné summity a užívateľské skupiny, kde zdieľa skúsenosti. Frekvencia aktualizácií je ~2× ročne, bezpečnostné updaty vychádzajú priebežne. . Dostupnosť školení a certifikácií je veľmi vysoká:

- **Komunitná podpora**: Veľká a aktívna komunita s príspevkami od spoločností ako Red Hat, Canonical a SUSE. Podpora je dostupná cez mailing listy, fóra a lokálne používateľské skupiny (OpenStack Community). Konferencie ako OpenStack Summit zvyšujú angažovanosť komunity.
- **Dokumentácia**: Rozsiahla dokumentácia na docs.openstack.org pokrýva inštaláciu, správu, vývoj a riešenie problémov. Zahŕňa príručky pre rôzne distribúcie Linuxu a scenáre nasadenia (OpenStack Installation Guide).
- **Frekvencia aktualizácií**: Nové verzie vychádzajú každých šesť mesiacov, s názvami podľa texaských miest. Staršie verzie majú predĺženú údržbu (Extended Maintenance), čo zaisťuje dlhodobú podporu (OpenStack Releases).
- **Podpora a školenia**: Dostupné sú certifikácie (napr. Certified OpenStack Administrator) a profesionálna podpora od firiem ako Red Hat a Canonical (Red Hat OpenStack).

Odôvodnenie: Aktívna a široká komunita, bohatá a neustále aktualizovaná dokumentácia, pravidelné a časté aktualizácie. Dostupná rozsiahla profesionálna podpora a certifikácie od viacerých významných komerčných subjektov, čo zaručuje dlhodobú udržateľnosť projektu SOC. Platforma má dlhodobo udržateľný vývoj a podporu, čo je kľúčové pre kritické nasadenia.

Hodnotenie: 5 b

3.3.2.2 CloudStack

Popis: CloudStack má aktívnu, no menšiu komunitu v porovnaní s OpenStackom. Je to projekt Apache Software Foundation, čo zabezpečuje určitú úroveň transparentnosti a otvorený riadiaci model. Vydávanie nových verzií je pomerne pravidelné, aj keď frekvencia môže byť o niečo nižšia ako pri OpenStacku. Dokumentácia je dobrá a dostatočná pre bežné

nasadenia a správu, ale nemusí byť tak podrobná pre okrajové prípady alebo komplexné integrácie. Existujú niektorí komerční dodávatelia (napr. ShapeBlue, StorPool), ktorí poskytujú podporu, školenia a konzultácie, ale ich ekosystém je menej rozsiahly. Celkovo je podpora a komunita stabilná, ale nie tak masívna a s takými zdrojmi ako u OpenStacku alebo Kubernetes:

- **Komunitná podpora**: Menšia komunita ako OpenStack, ale stále aktívna, s príspevkami od firiem ako Citrix a ShapeBlue. Podpora je dostupná cez mailing listy a fóra (CloudStack Mailing Lists). Diskusie na platformách ako Reddit naznačujú, že komunita je angažovaná, hoci menšia (CloudStack Reddit Discussion).
- **Dokumentácia**: Komplexná dokumentácia na docs.cloudstack.apache.org pokrýva inštaláciu, správu a pokročilé konfigurácie. Je menej rozsiahla ako OpenStack, ale dostatočná pre väčšinu použití (CloudStack Documentation).
- **Frekvencia aktualizácií**: Pravidelné vydania, s najnovšou LTS verziou 4.20.1.0 v roku 2023. Aktualizácie sú menej časté ako u OpenStack, ale stále spoľahlivé (CloudStack Releases).
- **Podpora a školenia**: Certifikácia ACCEL od LPI-JAPAN je dostupná, ale profesionálna podpora je obmedzená na menší počet poskytovateľov ([ACCEL Certification]).

Odôvodnenie: CoudStack si udržuje aktívnu, no pomerne úzko zameranú komunitu. Počet prispievateľov a používateľov nie je taký veľký ako pri OpenStacku, avšak medzi referenčných užívateľov patria aj významné spoločnosti (Telefónica, BT, univerzity), čo svedčí o dôvere v projekt. Dokumentácia je hodnotená ako dostačujúca pre bežné nasadenia, hoci nie je tak rozsiahla ako pri OpenStacku. Výhodou je Apache model – transparentné riadenie projektu a komunita orientovaná na stabilitu. Dostupnosť profesionálnej podpory (napr. ShapeBlue ponúka 24/7 support) zvyšuje atraktivitu pre firmy, ktoré nemajú vlastný tím expertov. Celkovo má CloudStack v oblasti komunity a podpory silné postavenie,

Hodnotenie: 3 b

3.3.2.3 Proxmox VE

Popis: Proxmox VE je podporovaný aktívnou komunitou administrátorov, nadšencov a malých firiem. Oficiálne fórum Proxmox má desaťtisíce príspevkov, kde vývojári aj skúsení užívatelia radia nováčikom (fórum je považované za výborný zdroj riešení – *„great place to find an answer“*). Dokumentácia projektu je dostupná na wiki (pve.proxmox.com) a zahŕňa užívateľské príručky, často kladené otázky aj postupy pre pokročilé nastavenia. Proxmox Server Solutions GmbH (tvorca projektu) ponúka platené enterprise predplatné, ktoré garantuje prístup k stabilným repozitárom a technickej podpore. Komunita vydáva veľa tutoriálov (YouTube, blogy) zameraných napr. na clustering, zálohovanie, integráciu s ZFS atď. Aktualizácie Proxmox VE vychádzajú pravidelne – hlavné verzie približne raz ročne, minor updaty priebežne; upgrade proces je dobre zdokumentovaný:

- **Komunitná podpora**: Aktívna komunita s podporou cez fórum (Proxmox Support Forum). Komerčná podpora je dostupná cez predplatné, čo zvyšuje spoľahlivosť (Proxmox Subscription).
- **Dokumentácia**: Podrobná dokumentácia na pve.proxmox.com pokrýva inštaláciu, správu a pokročilé funkcie. Je užívateľsky prívetivá a pravidelne aktualizovaná (Proxmox Documentation).
- **Frekvencia aktualizácií**: Nové verzie vychádzajú každých pár mesiacov, s vydaním 8.2 v roku 2024. Aktualizácie sú dostupné cez repozitáre (Proxmox Roadmap).
- **Podpora a školenia**: Komerčné školenia a podpora sú dostupné, s pozitívnymi recenziami od používateľov ([Proxmox Reviews]).

- **Skóre**: 5 bodov – Aktívna komunita, bohatá dokumentácia, pravidelné aktualizácie a profesionálna podpora

Odôvodnenie: V rámci open-source projektov má Proxmox silnú a angažovanú komunitu, ťažiskom ktorej je oficiálne fórum a početné príspevky na blogoch či Reddite od používateľov v homelab a SMB sfére. Keďže ide o populárne riešenie pre menšie nasadenia, komunita je veľmi prakticky orientovaná – riešenia bežných problémov sú často vyhľadateľné v diskusiách. Dokumentácia pokrýva kľúčové témy, hoci v porovnaní s veľkými cloud platformami je menej obsiahla (často sa spolieha na znalosti Linuxu, na ktorom Proxmox stojí). Dostupnosť platenej podpory priamo od tvorcov je veľkým plus pre nasadenie v podnikovej sfére – za relatívne nízky poplatok ročne je možné mať prístup k expertom. Celkovo Proxmox dosahuje vysoké hodnotenie, pričom jediným dôvodom pre nie plných 5 bodov je menší rozsah projektu (a teda komunity) oproti gigantickým cloudovým riešeniam

Hodnotenie: 5 b

3.3.2.4 Kubernetes

Popis: Kubernetes má jednu z najdynamickejších a najväčších open-source komunít v IT svete, ktorá je riadená prostredníctvom Cloud Native Computing Foundation (CNCF). Projekt má obrovské množstvo prispievateľov od spoločností ako Google, Red Hat, Microsoft, ako aj nezávislých vývojárov. Vydávanie nových verzií je mimoriadne časté (približne každé 3-4 mesiace), čo prináša rýchly vývoj funkcií a bezpečnostných opráv. Dokumentácia je mimoriadne rozsiahla a neustále aktualizovaná, aj keď pre nováčikov môže byť jej komplexnosť výzvou. Existuje nespočetné množstvo kníh, kurzov, certifikácií (napr. CKA, CKAD, CKS) a konzultačných služieb od prakticky všetkých veľkých IT firiem. Celý ekosystém okolo Kubernetes je obrovský a neustále rastie.

Odôvodnenie: Najaktívnejšia a najväčšia komunita v oblasti cloud-native. Disponuje extrémne bohatou dokumentáciou, veľmi častými aktualizáciami a širokou ponukou profesionálnej podpory, školení a certifikácií od vedúcich IT spoločností, čo zaručuje dlhodobú udržateľnosť a robustnú podporu pre SOC projekt.

Hodnotenie: 5 b

3.4 BEZPEČNOSTNÉ KRITÉRIÁ

3.4.1 K9: Autentifikácia a riadenie prístupov (váha: 5)

Odôvodnenie: Platforma by mala obsahovať zabudované bezpečnostné prvky potrebné na ochranu infraštruktúry a dát v prostredí SOC. Sem patrí napríklad podpora pre viacúrovňovú autentifikáciu a autorizáciu užívateľov, možnosť šifrovania dát (v úložiskách, prenosoch), logovanie a auditovanie udalostí či automatizované bezpečnostné politiky. Pri open-source riešení posudzujeme, aké bezpečnostné moduly ponúka a či spĺňa moderné bezpečnostné postupy (pravidelné bezpečnostné aktualizácie, možnosti hardeningu).

Vzhľadom na povahu projektu (kyberbezpečnosť) ide o **kritické** kritérium s váhou **5**. Platforma, ktorá by nevedela zabezpečiť dáta a systémy na adekvátnej úrovni, nemôže byť považovaná za vhodnú.

Odporúčanie pre hodnotenie:

- 5 bodov: Pokročilá viacúrovňová autentifikácia, robustné riadenie prístupov, úplná podpora logovania a auditu.

- 3 body: Základná autentifikácia a riadenie prístupov, limitované možnosti auditovania.
- 1 bod: Slabá bezpečnosť autentifikácie, minimálne možnosti auditu.

3.4.1.1 OpenStack

Popis: OpenStack obsahuje robustný identitný modul **Keystone**, ktorý zabezpečuje autentifikáciu užívateľov a autorizáciu ich prístupu k zdrojom. Keystone podporuje viacúrovňové overovanie – základom je tokenový mechanizmus (užívateľ sa prihlási menom/heslom alebo iným mechanizmom a získa časovo obmedzený token pre volanie služieb). Možná je integrácia s externými adresármi ako LDAP/Active Directory [61], a taktiež federovaná autentifikácia (SAML, OpenID Connect) pre jednotné prihlásenie. Keystone umožňuje nastaviť viacfaktorovú autentifikáciu (MFA) – napr. kombinácia hesla a jednorazového TOTP kódu [62]. Riadenie prístupu je riešené pomocou role-based access control (RBAC): administrátori definujú roly a priraďujú ich užívateľom v projektoch. V OpenStacku sú preddefinované roly (admin, member, reader) a od Wallaby release je k dispozícii unifikovaný default RBAC policy pre konzistentné oprávnenia naprieč službami.

- **Viacúrovňová autentifikácia a autorizácia**: Podporuje integráciu s externými systémami ako LDAP/AD, s RBAC založeným na rolách, skupinách a užívateľoch
- **Podpora MFA**: Môže byť rozšírená cez pluginy alebo externé služby, hoci nie je natívne podporovaná.
- **Šifrovanie dát**: Podporuje TLS pre dáta v prenose a môže integrovať šifrované úložiská pre dáta v pokoji.
- **Logovanie a auditovanie**: Komplexné logovanie udalostí autentifikácie a auditovania, s podporou nástrojov ako ELK stack.
- **Automatizované bezpečnostné politiky**: Umožňuje definovať politiky pre riadenie prístupov, ktoré môžu byť automatizované.
- **Moderné bezpečnostné postupy**: Pravidelné aktualizácie a sprievodcovia pre hardening .
- **RBAC**: Plne implementované cez Keystone, umožňujúce jemnú kontrolu prístupov.

Odôvodnenie: Poskytuje pokročilú viacúrovňovú autentifikáciu a robustné riadenie prístupov s detailným RBAC. Ponúka úplnú podporu pre logovanie, auditovanie a šifrovanie dát v kľude a pri prenose, spĺňa moderné bezpečnostné postupy a je aktívne udržiavaný z hľadiska bezpečnosti.

Hodnotenie: 5 b

3.4.1.2 CloudStack

Popis: CloudStack poskytuje vstavaný mechanizmus riadenia prístupu postavený na účtoch, doménach a roliach. Systém rozlišuje viacero úrovní administrátorov: root admin (celý cloud), domain-admin (správca konkrétnej domény/tenantu) a bežní užívatelia [63]. Pre každý účet je možné definovať viac užívateľov, ktorí zdieľajú zdroje. Autentifikácia prebieha štandardne cez meno/heslo, pričom CloudStack podporuje integráciu s externými systémami: umožňuje pripojenie na LDAP/AD pre overovanie užívateľov a taktiež konfiguráciu Single Sign-On cez SAML 2.0 [63]. Od verzie 4.11 CloudStack priniesol dynamické roly – administrátor môže vytvárať vlastné roly s určitou sadou oprávnení (preddefinované sú roly Admin, User, DomainAdmin). Dvojfaktorová autentifikácia je podporovaná voliteľným pluginom – dá sa zapnúť TOTP generovanie kódov pre prihlásenie do GUI [63]. Apache CloudStack ponúka nasledujúce vlastnosti:

- **Viacúrovňová autentifikácia a autorizácia**: Podporuje API kľúče, LDAP, SAML a OAuth2, s hierarchickým riadením prístupov cez role, účty, užívateľov a domény

- **Podpora MFA**: Explicitne podporuje 2FA a OAuth2, čo zvyšuje bezpečnosť.
- **Šifrovanie dát**: Podporuje TLS, šifrovanie dát v pokoji závisí od podkladového úložiska.
- **Logovanie a auditovanie**: Poskytuje logovanie, ale auditovanie môže vyžadovať dodatočnú konfiguráciu.
- **Automatizované bezpečnostné politiky**: Umožňuje definovať politiky cez API a konzolu, s obmedzenou automatizáciou.
- **Moderné bezpečnostné postupy**: Pravidelné aktualizácie, podporuje moderné štandardy ako SAML a OAuth2.
- **RBAC**: Implementuje RBAC cez role a domény, umožňujúce jemnú kontrolu.

Odôvodnenie: Bezpečnostné funkcie CloudStacku pokrývajú väčšinu potrieb multitenant cloudu. Oddelenie tenantov je vynútené logicky (domény) a všetky API volania kontrolujú, či volajúci má prístup k objektu. Možnosť integrovať LDAP a SAML umožňuje organizáciám použiť existujúce identity manažment riešenia na prihlasovanie do CloudStacku. Hoci CloudStack nemá tak granule nastavenia ako OpenStack policy engine, jeho model rolí je dostatočný pre bežné scenáre (bežný užívateľ nemôže ovplyvniť iného, domain-admin vidí iba svoju doménu). Pozitívom je nativna podpora 2FA pre UI, čím sa zvyšuje bezpečnosť prístupu pre správcovské účty [63]. Celkovo CloudStack ponúka robustnú, hoci o trochu jednoduchšiu správu prístupov v porovnaní s OpenStack – to môže byť výhodou z hľadiska prehľadnosti. Bezpečnostné best practices (unikátne API kľúče pre užívateľov, možnosť rotácie kľúčov) sú v CloudStacku implementované. Ponúka možnosti logovania, ale auditovanie môže byť limitované. Základné šifrovanie je prítomné.

Hodnotenie: 4 b

3.4.1.3 Proxmox VE

Popis: Proxmox VE sa primárne zameriava na správu virtualizácie na úrovni hosta. Poskytuje robustné autentifikačné mechanizmy vrátane interného Proxmox databázového systému, LDAP/AD, PAM a Realm (pre rôzne zdroje autentifikácie). Riadenie prístupov je založené na granulárnom RBAC, kde možno definovať roly a oprávnenia na úrovni VM, úložiska, siete atď. Všetky administrátorské akcie sú logované a dostupné prostredníctvom GUI alebo CLI. Proxmox podporuje šifrovanie diskov VM (cez LUKS) a zabezpečenú komunikáciu (SSH, HTTPS pre web GUI). Je to však systém zameraný na infraštruktúru, nie na cloudovú správu identít pre koncových používateľov služieb. Chýbajú mu pokročilé funkcie ako self-service portály pre užívateľov s viacúrovňovou autentifikáciou alebo automatizované bezpečnostné politiky aplikované na cloudovú službu ako celok. Hoci je samotný Proxmox veľmi bezpečný a pravidelne aktualizovaný, pre potreby komplexného SOC riešenia s multitenantným prístupom by bolo potrebné vybudovať ďalšie vrstvy IAM nad ním:

- **Viacúrovňová autentifikácia a autorizácia**: Podporuje viaceré zdroje autentifikácie, vrátane PAM, LDAP, AD a OpenID Connect, s flexibilným riadením prístupov
- **Podpora MFA**: Môže byť integrovaná cez externé zdroje, ale nie je natívne podporovaná.
- **Šifrovanie dát**: Podporuje šifrovanie pre VM a kontajnery, s integráciou šifrovaných úložísk.
- **Logovanie a auditovanie**: Poskytuje logovanie, auditovanie môže vyžadovať dodatočné nástroje.
- **Automatizované bezpečnostné politiky**: Umožňuje definovať povolenia a ACL pre skupiny, s obmedzenou automatizáciou.
- **Moderné bezpečnostné postupy**: Pravidelné aktualizácie, podporuje moderné štandardy ako OpenID Connect.

- **RBAC**: Implementuje RBAC cez povolenia a skupiny, umožňujúce jemnú kontrolu.

Odôvodnenie: Proxmox nie je primárne navrhnutý pre veľký počet oddelených tenantov, obsahuje všetky dôležité mechanizmy na bezpečnú správu prístupov v rámci jednej organizácie. Integrácia s LDAP/AD dovolí využívať firemné účty pre login, čo zjednodušuje správu identít. Granularita RBAC v Proxmoxe je pre menšie prostredia dostatočná – administrátor vie nastaviť veľmi konkrétne povolenia (napr. právo rebootovať konkrétnu VM bez možnosti ju zmazať). Navyše podpora 2FA priamo v rozhraní Proxmoxu výrazne znižuje riziko neoprávneného prístupu, čo nie je v mnohých podobných nástrojoch bežný štandard. Obmedzením môže byť, že Proxmox nezabezpečuje hard multi-tenancy izoláciu – ak by viac rôznych zákazníkov malo zdieľať ten istý Proxmox cluster, administratívne účty by museli byť nastavované veľmi opatrne.

Hodnotenie: 4 b

3.4.1.4 Kubernetes

Popis: Kubernetes má veľmi pokročilý systém autentifikácie a riadenia prístupov. Autentifikácia je flexibilná a môže zahŕňať certifikáty, tokeny, LDAP/AD, OAuth 2.0 (cez externé integrácie). Autorizácia je založená na robustnom RBAC (Role-Based Access Control), ktorý umožňuje extrémne granulárne definovanie oprávnení na úrovni klastra, namespace-ov a jednotlivých zdrojov (pods, deployments, services atď.). Všetky interakcie s Kubernetes API sú logované do auditného logu, čo poskytuje podrobný prehľad o aktivitách. Podporuje šifrovanie etcd databázy, šifrovanie v kľude pre secrets a sieťovú segmentáciu (Network Policies). Je navrhnutý s dôrazom na bezpečnosť v cloud-native prostredí a má aktívnu komunitu, ktorá neustále vyvíja bezpečnostné nástroje (napr. nástroje pre policy enforcement ako Gatekeeper, Kyverno). Aj keď je jeho zameranie na kontajnery, bezpečnostné princípy sú vysoko relevantné pre ochranu dát v SOC:

- **Viacúrovňová autentifikácia a autorizácia**: Podporuje certifikáty, tokeny, OIDC a service account autentifikáciu, s integráciou externých poskytovateľov
- **Podpora MFA**: Môže byť integrovaná cez OIDC s poskytovateľmi podporujúcimi MFA.
- **Šifrovanie dát**: Podporuje TLS pre dáta v prenose a šifrovanie v pokoji pre etcd.
- **Logovanie a auditovanie**: Vstavané auditovanie a logovanie všetkých požiadaviek na API server.
- **Automatizované bezpečnostné politiky**: Používa RBAC a iné mechanizmy na vynucovanie politík.
- **Moderné bezpečnostné postupy**: Pravidelné bezpečnostné aktualizácie, podporuje Pod Security Policies a Network Policies.
- **RBAC**: Plne implementované RBAC, umožňujúce jemnú kontrolu prístupov.

Odôvodnenie: Poskytuje pokročilú a flexibilnú viacúrovňovú autentifikáciu a extrémne robustné, granulárne riadenie prístupov (RBAC). Má úplnú podporu pre detailné logovanie a auditovanie udalostí API. Aktívne podporuje šifrovanie a implementáciu automatizovaných bezpečnostných politík, čo z neho robí ideálnu voľbu pre kyberbezpečnostné prostredie

Hodnotenie: 5 b

3.4.2 K10: Izolácia tenantov a ochrana dát (váha: 5)

Odôvodnenie: V multitenantnom prostredí je nevyhnutné, aby platforma zaručila striktnú izoláciu medzi jednotlivými zákazníkmi (tenantmi) – ich virtuálne stroje, úložiská a siete musia byť oddelené tak, aby sa zabránilo neoprávnenému prístupu alebo úniku dát medzi tenantmi. Toto kritérium hodnotí architektúru platformy z hľadiska viacvrstvovej bezpečnosti:

oddelenie výpočtových zdrojov, sieťová segmentácia, mechanizmy kontroly prístupu a podobne. Miera izolácie dát medzi tenantmi je zásadným aspektom multitenantných systémov.

Akákoľvek slabina v izolácii by mohla viesť k bezpečnostným incidentom, preto tomuto kritériu priraďujeme **váhu 5** (veľmi vysoká).

Odporúčanie pre hodnotenie:

- **5 bodov:** Vysoká izolácia tenantov, úplná segmentácia zdrojov, pokročilé mechanizmy ochrany dát.
- **3 body:** Čiastočná izolácia s niektorými limitmi, menej robustné mechanizmy ochrany.
- **1 bod:** Slabá izolácia tenantov, riziko úniku dát, slabá kontrola prístupu.

3.4.2.1 OpenStack

Popis: OpenStack bol od základu navrhnutý pre multitenantné prostredia – každý užívateľ (resp. projekt/tenant) je oddelený na všetkých úrovniach. Virtuálne stroje rôznych projektov bežia na hypervízoroch izolovane prostredníctvom hypervízorovej vrstvy (KVM zabezpečuje pamäťovú aj procesorovú izoláciu VM). Sieťová izolácia je riešená v službe Neutron: tenanti majú vlastné virtuálne siete (VLAN, VXLAN) a medzi nimi nie je komunikácia, pokiaľ ju explicitne nepovolí cloud admin (napr. cez router s pravidlami) – štandardne teda jeden tenant nemôže „vidieť“ do siete iného. Úložiská a objemy (Cinder) sú priraďované vždy jednému projektu a prístup k nim je autentizovaný cez Keystone, takže jeden tenant nemá prístup k diskovým objemom druhého. Navyše OpenStack podporuje šifrovanie dát – napríklad Cinder umožňuje šifrovať volume na úrovni bloku (s kľúčmi uloženými v službe Barbican) a Glance vie uložiť QCOW2 obrazy so šifrovaním. Pre ochranu dát pri prenose poskytuje OpenStack end-to-end šifrovanie API komunikácie a možnosť nasadiť VPN pre tenantov (VPN-as-a-service). Vzhľadom na **SOCaaS (Security Operations Center as a Service)** nasadenie – ktoré vyžaduje multitenantné a bezpečnostne orientované prostredie – má OpenStack významné výhody. Už z návrhu počíta s multi-tenancy: izolácia nájomcov je zabezpečená na úrovni virtuálnych strojov (úplná softvérová izolácia VM), virtuálnych sietí (privátne siete, VLAN/VXLAN segregácia cez Neutron) aj úložiska (oddelené volumy a objekty). Každý tenant (projekt) má vlastné zdroje a identity spravované cez Keystone. **OpenStack je teda schopný poskytovať silnú izoláciu a bezpečnostné mechanizmy pre viacerých zákazníkov v jednom cloude** [36]:

- **Výpočtová izolácia**: Používa hypervizory ako KVM, ktoré zaisťujú silnú izoláciu virtuálnych strojov (VM) medzi tenantmi (projektmi). Každý projekt má svoje vlastné VM, ktoré sú oddelené od ostatných (OpenStack Multi-Tenant Isolation).
- **Sieťová segmentácia**: Služba Neutron podporuje izolované siete tenantov cez VLAN, VXLAN a bezpečnostné skupiny, čím zaisťuje oddelenie sieťového prenosu (OpenStack Networking Security).
- **Úložisková izolácia**: Služby Cinder (blokové úložisko) a Swift (objektové úložisko) zaisťujú oddelenie dát tenantov pomocou projektových účtov a kontajnerov (OpenStack Security Guide).
- **Riadenie prístupu**: Služba Keystone poskytuje role-based access control (RBAC) a integráciu s LDAP/AD, čím zaisťuje, že používatelia majú prístup iba k zdrojom svojho projektu.
- **Ochrana dát**: Podporuje šifrovanie dát v prenose (TLS) a v pokoji (napr. šifrované zväzky v Cinder). Dokumentácia popisuje postupy pre hardening a ochranu dát (OpenStack Hardening Guide).
- **Pokročilé mechanizmy**: Zahŕňa bezpečnostné skupiny, firewall pravidlá a integráciu s externými bezpečnostnými nástrojmi, čím minimalizuje riziko úniku dát.

- **Známe zraniteľnosti**: OpenStack má procesy pre správu zraniteľností cez OpenStack Security Advisories (OSSA) a Vulnerability Management Team (VMT), ktoré rýchlo riešia bezpečnostné problémy (OpenStack Security Portal).

Odôvodnenie: Vysoká izolácia tenantov je jadrom OpenStacku. Poskytuje úplnú segmentáciu výpočtových zdrojov, sietí a úložísk, spolu s pokročilými mechanizmami kontroly prístupu (RBAC) a šifrovaním, čo minimalizuje riziko úniku dát medzi tenantmi.

Hodnotenie: 5 b

3.4.2.2 CloudStack

Popis: CloudStack takisto poskytuje viacvrstvovú izoláciu medzi tenantmi (v CloudStack terminológii účtami resp. doménami). Na sieťovej úrovni využíva CloudStack v "advanced" zónach samostatné VLAN pre siete každého tenanta, prípadne VXLAN pre overlay siete – zákazníci tak majú oddelený broadcast domain a adresný rozsah. V "basic" zónach (bez VLAN) zas CloudStack oddeľuje VM cez bezpečnostné skupiny (L3 firewall pravidlá podobné AWS EC2 klasickému módu). Úložiská: diskové objemy a šablóny sú vždy asociované s konkrétnym účtom; pre každý účet vie CloudStack obmedziť prístup ku storage zdrojom len na jeho VM. Admin má možnosť definovať **zóny** a **domény**, čím sa dajú zákazníci segregovať aj geograficky či infraštruktúrne. Ochrana dát je podporená funkcionalitami ako šifrovanie diskov na úrovni hypervízora KVM (od verzie 4.11 je dostupná podpora šifrovania LUKS pre volume). CloudStack má zabudovaný systém **snapshotov** a **záloh**, pričom tie sú tiež viazané na účet – jeden tenant nevidí ani nemôže obnoviť snapshot iného. Mechanizmy high availability (napr. automatické reštarty VM pri výpadku hostiteľa) prispievajú k ochrane integrity dát tým, že minimalizujú riziko pádu celého systému pre jedného klienta:

- **Výpočtová izolácia**: Používa hypervizory ako KVM a Xen, ktoré zaisťujú izoláciu VM medzi účtami alebo doménami (CloudStack Networking).
- **Sieťová segmentácia**: Podporuje izolované siete a bezpečnostné skupiny, ktoré zaisťujú oddelenie sieťového prenosu tenantov pomocou VLAN a virtuálnych sietí (CloudStack Security Groups).
- **Úložisková izolácia**: Spravuje úložné fondy a zväzky, ktoré sú priradené konkrétnym účtom, čím zaisťuje oddelenie dát (CloudStack Administration).
- **Riadenie prístupu**: Implementuje RBAC cez účty a domény, s podporou API kľúčov, LDAP a SAML, čím zaisťuje kontrolu prístupu.
- **Ochrana dát**: Podporuje TLS pre dáta v prenose a môže integrovať šifrované úložiská. Dokumentácia popisuje bezpečnostné postupy (CloudStack Security).
- **Pokročilé mechanizmy**: Ponúka sieťové ACL a bezpečnostné skupiny, hoci menej pokročilé ako OpenStack.
- **Známe zraniteľnosti**: Niektoré CVEs: CloudStack má zraniteľnosti, ako CVE-2024-42062, ktoré boli opravené v novších verziách (CloudStack Security Advisory).

Odôvodnenie: Z hľadiska izolácie poskytuje CloudStack porovnateľný stupeň oddelenia ako OpenStack v bežných scenároch. Každý zákazník funguje v rámci svojej domény, so separátnymi sieťami a zdrojmi. CloudStack priamo zabraňuje prideleniu jednej verejnej IP dvom rôznym účtom alebo pripojeniu VM do siete, ku ktorej nemajú oprávnenie. Sieťová segmentácia cez VLAN/VXLAN je spoľahlivá metóda využívaná aj v iných cloudoch, čím je garantované, že napríklad broadcast či ARP prevádzka sa nedostane mimo tenant. Ochrana dát šifrovaním nie je v CloudStacku tak komplexne rozvinutá ako v OpenStacku (chýba ekvivalent služby Barbican na centrálne manažovanie kľúčov), no integráciou s hypervízorovými nástrojmi dokáže aj CloudStack zabezpečiť šifrované disky. Z auditov nasadení (Telia, BT a iní telco operátori používajú CloudStack) vyplýva, že platforma je

dostatočne bezpečná pre multitenentné prostredie. Jedinou potenciálnou slabinou môže byť menší ekosystém bezpečnostných rozšírení – napríklad menej vstavaných možností pre IDS/IPS integráciu. Napriek tomu CloudStack v kritériu izolácie tenantov a ochrany dát takmer nezaostáva za OpenStackom.

Hodnotenie: 5 b

3.4.2.3 Proxmox VE

Popis: Proroxmox VE nie je primárne navrhnutý pre viacnájomníkov z rôznych organizácií, ale skôr pre centralizovanú virtualizáciu v rámci jedného administratívneho okruhu. Izolácia medzi VM je zabezpečená na úrovni hypervízora KVM – každá VM beží v samostatnom procese QEMU s vlastným izolovaným pamäťovým priestorom. Pre kontejnery LXC Proxmox využíva izoláciu cez Linux namespaces a cgroups; tie poskytujú oddelenie, avšak zdieľajú isté časti kernelu s hostiteľom. Čo sa týka sieťovej izolácie, Proxmox neponúka multi-tenant SDN out-of-the-box – administrátor však môže využívať VLAN tagging na oddeľovanie sietí rôznych skupín VM, alebo nastaviť úplne samostatné fyzické siete pre rôznych užívateľov. Ochrana dát v Proxmox prostredí stojí najmä na zálohovaní a replikačných mechanizmoch: Proxmox umožňuje plánované zálohy VM (napr. do Proxmox Backup Servera) a replikácie VM medzi uzlami clusteru, čo chráni dáta pred zlyhaním HW. Šifrovanie diskov VM nie je spravované priamo Proxmoxom (je ponechané na hosťujúci OS vo vnútri VM, prípadne na úrovni úložiska – ZFS encryption). Pri použití Ceph storage v rámci Proxmox clusteru je možné využiť natívne Ceph mechanizmy ochrany dát:

- **Výpočtová izolácia**: Používa KVM a LXC, ktoré zaisťujú izoláciu VM a kontajnerov, ale multitenancia vyžaduje dodatočné konfigurácie (Proxmox Multi-Tenancy).
- **Sieťová segmentácia**: Podporuje VLAN a OpenVSwitch pre sieťovú izoláciu, ale konfigurácia je manuálna a zložitá (Proxmox OVS Networking).
- **Úložisková izolácia**: Úložiská môžu byť priradené konkrétnym VM, ale nie sú natívne navrhnuté pre multitenancu.
- **Riadenie prístupu**: Podporuje RBAC a integráciu s LDAP/AD, ale nie je optimalizované pre multitenantné prostredia.
- **Ochrana dát**: Podporuje šifrovanie VM a záloh, ale pokročilé bezpečnostné funkcie vyžadujú externé nástroje.
- **Pokročilé mechanizmy**: Nástroje ako MultiPortal umožňujú multitenancu, ale nie sú natívne (MultiPortal).
- **Známe zraniteľnosti**: CVEs ako SSRF boli opravené, ale multitenantné zraniteľnosti nie sú špecificky zdokumentované (Proxmox CVEs).

Odôvodnenie: Má slabú izoláciu tenantov na cloudovej úrovni, pretože nie je primárne navrhnutý ako multitenantná cloudová platforma. Poskytuje izoláciu VM na úrovni hypervízora, ale chýbajú mu integrované mechanizmy pre automatizovanú segmentáciu a kontrolu prístupu medzi rôznymi tenantmi bez rozsiahlych externých nástrojov a manuálneho úsilia. Pokiaľ by však Proxmox mal slúžiť ako verejný cloud pre neznámych zákazníkov, chýbajúce integrované bariéry (napr. oddelené virtuálne siete na klik pre každého tenanta, automatizované firewall pravidlá medzi tenantmi) predstavujú riziko nesprávnej konfigurácie a potenciálneho prieniku.

Hodnotenie: 1,5 b

3.4.2.4 Kubernetes

Popis: Kubernetes je zameraný na izoláciu kontajnerov a mikroslužieb, čo je pre multitenantné prostredie kritické, aj keď primárne nie pre IaaS virtuálne stroje. Poskytuje silné mechanizmy izolácie prostredníctvom Namespace-ov (logické oddelenie zdrojov), Network

Policies (granulárna sieťová segmentácia medzi podmi/namespace-mi), Resource Quotas (obmedzenie spotreby zdrojov), Security Contexts a Pod Security Standards (zabezpečenie kontajnerov). RBAC (Role-Based Access Control) je extrémne robustný a umožňuje veľmi detailné riadenie prístupu k API aj k zdrojom v rámci klastra. Všetky interakcie sú auditované. Hoci izolácia VM prostredníctvom KubeVirt je možná, natívne sa izolácia Kubernetes týka kontajnerov. Avšak princípy a nástroje pre izoláciu, ktoré Kubernetes poskytuje, sú vysoko prenosné a aplikovateľné aj na bezpečnostné aspekty multitenantnosti, ak sa riešenie postaví na kontajneroch alebo VM pod Kubernetesom:

- **Výpočtová izolácia**: Kontajnery zdieľajú kernel, ale názvové priestory a nástroje ako GKE Sandbox zaisťujú izoláciu (Kubernetes Multi-Tenancy).
- **Sieťová segmentácia**: Sieťové politiky obmedzujú prenos medzi názvovými priestormi, čím zaisťujú izoláciu (GKE Multi-Tenancy).
- **Úložisková izolácia**: Persistent Volumes a Claims sú viazané na názvové priestory, čím zaisťujú oddelenie dát.
- **Riadenie prístupu**: RBAC a Pod Security Standards zaisťujú jemnú kontrolu prístupu (Kubernetes Security).
- **Ochrana dát**: Podporuje šifrovanie dát v prenose (TLS) a v pokoji (etcd), s pokročilými bezpečnostnými politikami.
- **Pokročilé mechanizmy**: Pod Security Policies, admission controllers a nástroje ako GKE Sandbox zvyšujú bezpečnosť (GKE Sandbox).
- **Známe zraniteľnosti**: Zraniteľnosti sú riešené rýchlo, s dôrazom na správnu konfiguráciu (Kubernetes Vulnerabilities).

Odôvodnenie: Kubernetes poskytuje vysokú izoláciu zdrojov (prostredníctvom Namespace-ov, Network Policies, RBAC) a pokročilé mechanizmy ochrany dát pre kontajnerové prostredia, ktoré sú rovnako kritické pre multitenantnosť. Zdieľanie kernelu medzi kontajnermi však nesie určité riziká – hoci zraniteľnosti únikov z kontejnerov sú zriedkavé, existujú.

Hodnotenie: 4,5 b

3.4.3 K11: Sieťová/monitorovacia integrácia (váha: 3)

Odôvodnenie: Odôvodnenie: Hodnotí mieru integrácie s nástrojmi pre monitoring a bezpečnostné dohľadové systémy – najmä SIEM, IDS/IPS (napr. Suricata, Snort), ako aj metrické/logovacie nástroje ako Ceilometer, Grafana, Prometheus. Zohľadňuje tiež schopnosť platformy exportovať bezpečnostné udalosti, podporu API/webhookov a možnosti integrácie s externými bezpečnostnými prvkami. Kritérium dopĺňa hodnotenie autentifikácie a izolácie o schopnosť platformy byť súčasťou aktívneho bezpečnostného prostredia.

Odporúčanie pre hodnotenie:

- 5 bodov: Úplná integrácia s nástrojmi (SIEM, IDS/IPS, Grafana, Prometheus), jednoduchá konfigurácia.
- 3 body: Čiastočná integrácia, niektoré dôležité nástroje sú podporované obmedzene.
- 1 bod: Veľmi obmedzená integrácia, problémy s prepojením nástrojov a exportom udalostí.

3.4.3.1 OpenStack

Popis: OpenStack poskytuje bohaté možnosti integrácie s monitorovacími a bezpečnostnými nástrojmi. Pre monitoring výkonu a zdrojov má komponent Ceilometer (Telemetry), ktorý zbiera metriky z rôznych služieb (CPU zaťaženie VM, využitie diskov,

sieťová prevádzka) a môže ich ukladať či posielať ďalej. Na Ceilometer nadväzuje Aodh (alarms) – umožňuje definovať alarmy na určité metriky, napr. upozorniť ak VM prekročí 90% CPU. Okrem toho existuje projekt Monasca, ktorý je kompletným monitorovacím riešením (time-series databáza, Grafana dashboardy) integrovaným do OpenStacku. OpenStack služby vedia publikovať udalosti a logy, ktoré môžu byť napojené na SIEM – napr. Keystone generuje audit eventy vo formáte CADF, použiteľné v bezpečnostných monitoroch. Na sieťovú integráciu OpenStack ponúka modul Tap-as-a-Service (TaaS) pre Neutron, ktorý vie zrkadliť sieťovú prevádzku z konkrétneho portu do analyzátora (IDS systému) v rámci cloudu. Ďalej je možné nasadiť virtuálne sieťové funkcie (VNF) ako firewall, IDS (napr. Suricata VM) a zapojiť ich do topológie sietí tenantov. OpenStack API umožňuje externým nástrojom (Zabbix, Prometheus) pravidelne vyžiadať stav zdrojov a štatistiky – mnohé moderné integrácie používajú priamo Prometheus OpenStack exporter, ktorý sprístupňuje metriky OpenStacku do systému Prometheus/Grafa:

- **SIEM integrácia**: Existuje špecifická aplikácia Splunk App for OpenStack Analytics, ktorá umožňuje monitorovanie a analýzu udalostí (OpenStack Analytics Tool — Crest Data). ELK stack (Elasticsearch, Logstash, Kibana) je tiež dobre integrovaný, s podrobnými sprievodcami na nastavenie (OpenStack Monitoring With ELK).
- **IDS/IPS**: Podporuje nasadenie IDS/IPS systémov na virtuálnych strojoch, s možnosťou použitia Tap-as-a-Service (TaaS) pre zrkadlenie portov v Neutron (Mirantis Documentation: IDS/IPS).
- **Metric/Logovacie nástroje**: Má vstavaný Ceilometer pre telemetriu, a integruje sa s Prometheus a Grafana, s dostupnými dashboardmi (Monitoring an Openstack deployment with Prometheus and Grafana).
- **Export udalostí a API**: Podporuje export bezpečnostných udalostí cez API a webhooky, s robustnou dokumentácio

Odôvodnenie: Úplná integrácia s nástrojmi pre monitoring (Ceilometer, Grafana, Prometheus) a bezpečnostnými systémami (SIEM, IDS/IPS). Poskytuje rozsiahle API a mechanizmy pre export bezpečnostných udalostí a flexibilnú konfiguráciu, čo je kľúčové pre aktívne bezpečnostné prostredie SOC. Komplexnosť samotného OpenStacku však môže vyžadovať určité úsilie pri počiatočnej konfigurácii týchto integrácií.

Hodnotenie: 5 b

3.4.3.2 CloudStack

Popis: CloudStack disponuje vstavaným Event a Alert systémom, ktorý zaznamenáva rôzne udalosti (napr. vytvorenie VM, zlyhanie hostiteľa) a dokáže o nich informovať administrátorov. Integrácia s externými monitorovacími nástrojmi je podporená cez **Event Notification Framework** – CloudStack vie publikovať všetky udalosti do AMQP message bus (napr. RabbitMQ) alebo Apache Kafka, odkiaľ ich môžu odoberať tretie systémy. Týmto spôsobom je možné prepojiť CloudStack s SIEM – každá významná udalosť (pridanie užívateľa, zastavenie VM, chyba hypervízora) sa objaví ako správa na bus-e, ktorú SIEM vyhodnotí. Pre výkonové monitorovanie CloudStack priamo ponúka možnosť SNMP trapov – má definované MIB pre štatistiky hostiteľov a VM, ktoré môže posielať do SNMP kompatibilného monitoringu. Ďalej, existujú komunitné pluginy/exportery, napríklad pre Zabbix (oficiálny template) alebo Prometheus, ktoré čítajú metriky cez CloudStack API a poskytujú ich na vizualizáciu . Na sieťovú integráciu CloudStack ponúka možnosť zapojenia fyzických zariadení: je možné ho nakonfigurovať tak, aby spolupracoval s externými firewallmi a loadbalancermi (napr. Palo Alto, Netscaler) – tieto zariadenia potom poskytujú pokročilý dohľad nad prevádzkou. CloudStack taktiež podporuje tzv. **Traffic Mirroring** cez definovanie policy na port profile (hoci táto funkcia nie je tak zdokumentovaná ako v OpenStack TaaS).

- **SIEM integrácia**: Nemá priame natívne integrácie s SIEM systémami, ale logy môžu byť exportované do SIEM systémov cez tretie strany, ako je ELK, s pluginmi ako Log View
- Dokumentácia naznačuje obmedzenú podporu pre pokročilé SIEM funkcie.
- **IDS/IPS**: Umožňuje nasadenie IDS/IPS systémov na virtuálnych strojoch, s konfiguráciou sieťového prenosu, ale nie je tak priamo podporované ako u OpenStack.
- **Metric/Logovacie nástroje**: Podporuje monitorovanie cez nástroje ako Zabbix a Zenoss, ae integrácia s Prometheus a Grafana môže vyžadovať dodatočné nastavenia.
- **Export udalostí a API**: Má API pre export logov, ale konfigurácia môže byť zložitejšia.

Odôvodnenie: V porovnaní s OpenStack Telemetry CloudStack nemá natívny komponent zhromažďujúci časové rady metrík – spolieha sa skôr na externé nástroje, ktorým poskytne potrebné dáta. To je mierne mínus, ak by sme očakávali plne integrované grafy a alerty priamo v platforme (CloudStack UI zobrazuje len základné štatistiky o využití). Čo sa týka bezpečnostnej integrácie, CloudStack neponúka špecifické moduly pre IDS, ale umožňuje nasadiť IDS ako súčasť virtuálnej infraštruktúry a využiť eventy/alerty pre reakciu – napríklad alert o prekročení prahu CPU na VM môže byť indikátorom útoku a cez webhook by mohol spustiť skript na škálovanie. Celkovo CloudStack napĺňa požiadavky na prepojenie s dohľadom slušne, aj keď nie tak komplexne out-of-the-box ako niektorí konkurenti.

Hodnotenie: 3,5 b

3.4.3.3 Proxmox VE

Popis: Proxmox VE ponúka vstavané monitorovanie základných metrík a syslog pre logovanie udalostí, avšak nemá priamo integrované prepojenie na externé dohľadové systémy. Administrátorské web rozhranie zobrazuje aktuálne vyťaženie CPU, pamäte, IO aj sieťový traffic pre každý node a VM – tieto dáta sú uchovávané v RRD databázach a dajú sa exportovať. Pre pokročilejší monitoring sa v praxi využívajú externé nástroje: Proxmox má dostupný Prometheus exporter (open-source projekt) ktorý vie zberať dáta z Proxmox API a sprístupniť ich Prometheu. Podobne existujú šablóny pre Zabbix a integrácie pre Check_MK/Icinga, ktoré veľa správcov nasadzuje na monitoring Proxmox clusterov. Čo sa týka sieťovej integrácie, Proxmox sám neobsahuje virtuálne siete s pokročilými funkciami, takže integrácia s IDS/IPS spočíva na nasadení takých nástrojov ako VM či kontajner. Je však možné využiť fakt, že Proxmox beží na Linuxe – admin môže na hostiteľovi použiť linuxové nástroje (tcpdump, iptables mirror modul) ak potrebuje odchytávať prevádzku. Logy z Proxmoxu (vrátane audit logov – prihlásenia, operácie užívateľov) je možné presmerovať na externý syslog/SIEM server pre centralizovaný dohľad. Navyše Proxmox podporuje API hook skripty – pri určitých udalostiach (napr. štart/stop VM) vie spustiť vlastný skript, ktorým možno integrovať reakciu v externom systéme:

- SIEM integrácia: Môže byť integrovaný s SIEM nástrojmi ako Wazuh, s podrobnými sprievodcami na nastavenie . Logy môžu byť posielané do SIEM systémov cez syslog.
- IDS/IPS: Umožňuje spustenie IDS/IPS systémov ako virtuálnych strojov, s konfiguráciou zrkadlenia portov, ako je uvedené v sprievodcoch .
- Metric/Logovacie nástroje: Má vstavané monitorovanie, a integruje sa s Grafana a InfluxDB, s dostupnými dashboardmi . Prometheus je tiež podporovaný cez exportéry.
- Export udalostí a API: Podporuje export logov cez API a syslog, ale konfigurácia môže vyžadovať dodatočné kroky.

Odôvodnenie: V porovnaní s cloud platformami nemá Proxmox out-of-the-box prepracovaný monitoring alebo eventový systém, no jeho otvorenosť umožňuje adminom si potrebné prepojenia dorobiť. Napríklad nasadenie Prometheus+Grafana nad Proxmox clusterom je pomerne dobre zdokumentované komunitou a poskytne kompletný prehľad o zdrojoch (využitie CPU, uptime VM atď.). Pre menšie inštalácie to často ani nie je nutné – vstavané grafy v UI sú postačujúce pre bežný dohľad. Z pohľadu bezpečnostného monitoringu Proxmox spolieha na štandardné mechanizmy: logovanie všetkých akcií a možnosť ich centralizovať. Bezpečnostné incidenty (napr. zlyhanie live migrácie, výpadok uzla) Proxmox indikuje alarmom v UI a e-mail notifikáciou, ale neponúka integráciu s IDS. To odráža cieľovú oblasť Proxmoxu – nasadenie v dôveryhodnom prostredí, kde hlavným cieľom je jednoduchosť, nie komplexný bezpečnostný dohľad. V prípade potreby však nič nebráni zapojiť Proxmox do monitorovacieho systému podľa výberu organizácie. Zhrnutím, integráciu Proxmoxu s monitoringom hodnotíme ako uspokojivú, s pridanou námahou na strane administrátorov (nutné využiť externé nástroje a exportery).

Hodnotenie: 3,5 b

3.4.3.4 *Kubernetes*

Popis: Kubernetes je zameraný na observability (pozorovateľnosť) a má vynikajúce možnosti integrácie s nástrojmi pre monitoring, logovanie a trasovanie. Štandardom je integrácia s Prometheus (pre metriky), Grafana (pre vizualizáciu) a ELK stackom (Elasticsearch, Logstash, Kibana) alebo Loki/Fluentd (pre logy). Kubernetes Audit Logs poskytujú detailné záznamy všetkých interakcií s API serverom, ktoré sú kľúčové pre SIEM systémy. Integrácia s IDS/IPS je možná prostredníctvom nástrojov ako Falco (pre runtime bezpečnosť a detekciu hrozieb v kontajneroch), Calico pre sieťové politiky a iných CNI pluginov, ktoré umožňujú pokročilé sieťové funkcie a monitoring. Kubernetes má rozsiahle a konzistentné API, ktoré umožňuje programovú integráciu s prakticky akýmkoľvek externým systémom prostredníctvom operátorov a webhookov. Je ideálny pre nasadenie a integráciu kontajnerizovaných SIEM/IDS/IPS agentov:

- SIEM integrácia: Môže byť integrovaný s SIEM systémami cez logy a nástroje ako Wazuh, Falco, s podrobnými sprievodcami . Audit logy môžu byť exportované do SIEM systémov.
- IDS/IPS: Podporuje nasadenie IDS/IPS systémov ako podov, s nástrojmi ako Falco pre detekciu anomálií.
- Metric/Logovacie nástroje: Má štandardnú integráciu s Prometheus a Grafana, s mnohými predpripravenými dashboardmi .
- Export udalostí a API: Podporuje export udalostí cez API a webhooky, s robustnou dokumentáciou.

Odôvodnenie: Kubernetes patrí medzi najlepšie monitorovateľné platformy vôbec – už v základnej inštalácii sú dostupné metriky a obrovský výber integrácií v rámci CNCF projektu (Prometheus, Grafana, Loki pre logy, Jaeger pre tracing). Pre prevádzkovateľov to znamená, že dohľad nad Kubernetes klastrom a aplikáciami v ňom je možné realizovať jednotným spôsobom, a to aj vrátane VM vďaka KubeVirt. Keďže 80%+ organizácií využívajúcich Kubernetes nasadzuje aj cloud-native monitoring cncf.io, existuje množstvo znalostí a podpory pre nastavenie správnych metrík a alertov. Z bezpečnostného hľadiska Kubernetes nezaostáva: projekty ako Falco dokážu efektívne detegovať potenciálne útoky v runtime prostredí a network policy vedia obmedziť pohyb útočníka v prípade kompromitácie jedného podu. KubeVirt síce prináša klasické VM do prostredia Kubernetes, ale nekomplikuje integráciu – napríklad Falco vidí podozrivé aktivity aj v podoch KubeVirt (napr. ak by došlo k prelomeniu z VM do kontajnera). Navyše cloud-native prístup uľahčuje automatizovanú reakciu: na eventy z Kubernetes (či už monitorovacie alebo bezpečnostné) možno reagovať

pomocou Kubernetes Controllers alebo funkcií (napr. automaticky reštartovať pod, škálovať, zablokovať IP). Vzhľadom na tieto argumenty hodnotíme integrácie monitoringu a sieťového dohľadu v Kubernetes + KubeVirt ako výborné – komunita aj nástroje sú pripravené pokryť potreby produkčného nasadenia.

Hodnotenie: 5 b

4 HODNOTIACA TABUĽKA PRE VÝBER CLOUDOVEJ PLATFORMY

V nasledujúcej tabuľke sú uvedené výsledky multikriteriálneho hodnotenia štyroch kandidátnych cloudových platforiem (OpenStack, Apache CloudStack, Proxmox VE a Kubernetes s KubeVirt) podľa jedenástich výberových kritérií definovaných v dokumente V2. Každé kritérium má priradenú váhu vyjadrujúcu jeho dôležitosť (v rozsahu 1–5).

Hodnotenie bolo uskutočnené na základe odbornej analýzy dostupných zdrojov, dokumentácie (mnohé zdroje sa opakujú), prípadových štúdií a akademických publikácií. Pre každú platformu bola pre každé kritérium udelená známka na škále 1 až 5 bodov, ktorá bola následne vynásobená váhou daného kritéria.

Celkové skóre je teda váženým súčtom všetkých hodnotení. Platforma s najvyšším súčtom bodov predstavuje z pohľadu zvolených kritérií najvhodnejšie riešenie pre daný typ nasadenia.

		Openstack		CloudStack		Proxmox VE		Kubernetes	
Kritérium (Kx)	**Váha**	**Hodnotenie**	**Body celkom**	**Hodnotenie**	**Body celkom**	**Hodnotenie**	**Body celkom**	**Hodnotenie**	**Body celkom**
K1: Podpora IaaS funkcionality	5	5	25	4,5	22,5	3	15	1	5
K2: Modularita a architektúra	4	4,5	18	3	12	1	4	5	20
K3: Orchestrácia a automatizácia	3	5	15	4	12	3	9	5	15
K4: Výkonnosť a škálovateľnosť	5	5	25	5	25	3	15	5	25
K5: Spoľahlivosť a stabilita	4	5	20	5	20	4,5	18	5	20
K6: Štandardy a interoperabilita	4	5	20	3,5	14	3	12	4,5	18
K7: Náročnosť nasadenia a správy	3	1	3	4	12	5	15	1	3
K8: Dokumentácia a komunita	4	5	20	3,5	14	4	16	5	20
K9: Autentifikácia a prístupy	5	5	25	4	20	4	20	5	25
K10: Izolácia tenantov a ochrana dát	5	5	25	5	25	1,5	7,5	4,5	22,5
K11: Sieťová / monitorovacia integrácia	3	4,5	13,5	3,5	10,5	3,5	10,5	5	15
Súčet			209,5		187		142		188,5

Výsledkom vykonanej analýzy štyroch kandidátnych open-source platforiem (OpenStack, Apache CloudStack, Proxmox VE a Kubernetes / KubeVirt) na základe 11 definovaných kritérií je identifikácia ich silných a slabých stránok z pohľadu požiadaviek projektu SOCaaS (Security Operations Center as a Service) v multitenantnom prostredí.

Platformy boli hodnotené v oblastiach funkčnosti IaaS, modularity, škálovateľnosti, spoľahlivosti, interoperability, bezpečnosti a prevádzkových nárokov. Hodnotenie rešpektovalo vážený prístup, kde kritickým oblastiam (napr. bezpečnosť, výkonnosť, izolácia tenantov) bola priradená vyššia váha v súlade s cieľmi projektu.

OpenStack dosiahol najvyššie vážené skóre vo väčšine kategórií. Je to plnohodnotná IaaS platforma s rozvinutým ekosystémom komponentov (Compute, Networking, Storage), ktorá sa opiera o veľkú open-source komunitu, podporu štandardov a skúsenosti z rozsiahlych produkčných nasadení. Z pohľadu bezpečnosti poskytuje precízne mechanizmy autentifikácie, autorizácie, izolácie a šifrovania. Hoci jeho nasadenie je technicky náročné, návratnosť investície do robustného, modulárneho a škálovateľného riešenia prevyšuje úvodné náklady v prostredí s dlhodobým vývojom služieb.

Apache CloudStack ponúka nižšiu zložitosť nasadenia, veľmi dobrú spoľahlivosť a overenú prevádzkovú stabilitu. Predstavuje rozumne vyváženú alternatívu s nižšou komplexnosťou a prístupnejšou správou, pričom si zachováva schopnosť poskytovať základné IaaS funkcie a bezpečnostné prvky vhodné pre multitenantné prostredie. Zaostáva však v miere flexibility, rozšíriteľnosti a silnej podpory zo strany širšej komunity. V porovnaní skončil druhý.

Proxmox VE je jednoducho nasaditeľné virtualizačné riešenie, ideálne pre výučbové, testovacie alebo interné infraštruktúry. Jeho výhody spočívajú v rýchlej inštalácii, nízkych nárokoch na administráciu a stabilnej výkonnosti pre menšie prostredia. Proxmox však nie je navrhnutý pre rozsiahle multitenantné cloudové prevádzky s viac externými zákazníkmi s potrebou dôslednej izolácie tenantov, SLA podpory, pokročilej orchestracie alebo hybridného rozvoja služieb.

Kubernetes prináša koncept kontajnerizovaných a virtualizovaných workloadov pod jeden orchestrátor. Je mimoriadne silný v oblasti škálovania, automatizácie a integrácie monitorovacích nástrojov. Napriek rastúcej vyspelosti však v kontexte projektu naráža na určité limity: podpora VM je sekundárna, chýbajú tradičné IaaS mechanizmy (napr. komplexné storage API, samostatné tenant management, natívna sieťová virtualizácia na úrovni VM) a bezpečnostný model nie je primárne navrhnutý pre nepriateľský multi-tenancy bez silného dookola postaveného zabezpečenia. Do porovnania bol zaradený ako reprezentant kontajnerového prístupu.

5 ZÁVEREČNÉ ZHRNUTIE

Na základe vykonanej multikriteriálnej analýzy a posúdenia výsledkov jednotlivých platforiem odporúčame pre realizáciu cieľového riešenia SOCaaS nasadiť ako hlavnú technickú platformu OpenStack. OpenStack získal najvyššie hodnotenie odrážajúce úroveň technickej vyspelosti, flexibility, bezpečnosti a kompatibility so štandardmi. Je schopný poskytnúť robustnú IaaS infraštruktúru, ktorá umožňuje efektívne budovanie a prevádzku škálovateľného, multitenantného prostredia s dôrazom na bezpečnostné operácie. Riešenie je vhodné pre akademickú a výskumnú sféru, kde je dôležitá dlhodobá udržateľnosť, otvorenosť, nezávislosť od vendorov a možnosť zapojiť komunitné a výskumné aktivity. Ako alternatívu v prípade, že prevádzkové podmienky (personálne, časové alebo infraštruktúrne) neumožňujú okamžité nasadenie OpenStacku, odporúčame Apache CloudStack. Predstavuje rozumne zjednodušený kompromis medzi funkčnosťou a prevádzkovou nenáročnosťou, pričom si zachováva prijateľnú úroveň bezpečnosti a oddelenia tenantov. Proxmox VE odporúčame nasadzovať len pre obmedzené interné alebo výučbové scenáre bez kritických bezpečnostných alebo škálovacích požiadaviek. Kubernetes s KubeVirt predstavuje perspektívne rozšírenie v prípade prechodu na cloud-native model SOC služieb, avšak nie je vhodný ako primárne IaaS riešenie pre aktuálny cieľ projektu.

ZOZNAM ZDROJOV

1] „OpenStack,“ [Online]. Available: https://www.openstack.org/.

2] „Apache CloudStack,“ [Online]. Available: https://cloudstack.apache.org/.

3] A. Vogel, D. Greibler a C. Maron, „Private IaaS Clouds: A Comparative Analysis of OpenNebula, CloudStack and OpenStack,“ rev. *Conference: 2016 24th Euromicro International Conference on Parallel, Distributed, and Network-Based Processing (PDP)*, 2016.

4] „CloudStack vs OpenNebula comparison,“ [Online]. Available: https://www.peerspot.com/products/comparisons/cloudstack_vs_opennebula.

5] „Proxmox VE,“ [Online]. Available: https://www.proxmox.com/en/proxmox-ve.

6] „Kubernetes,“ [Online]. Available: https://kubernetes.io/.

7] „Open Nebula,“ [Online]. Available: https://opennebula.io/.

8] „oVirt,“ [Online]. Available: https://www.ovirt.org/.

9] „Eucalyptus,“ [Online]. Available: https://www.eucalyptus.cloud/.

10] „Harvester,“ [Online]. Available: https://harvesterhci.io/.

11] „Kubevirt,“ [Online]. Available: https://kubevirt.io/.

12] „The Most Widely Deployed Open Source Cloud Software in the World,“ [Online]. Available: https://www.openstack.org/software/project-navigator/openstack-components#openstack-services.

13] A. Rabcan, *Návrh architektúry hybrid-cloudového prostredia pre,* Žilina: UNIZA, 2020.

14] S. Hopko, *Problematika Cloud computing a jeho využitia,* UNIZA, 2015.

15] „OpenStack Architecture Guide,“ [Online]. Available: https://docs.openstack.org/arch-design/.

16] I. Voras, B. Mihaljevic a M. Orlic, „Criteria for Evaluation of Open Source Cloud Computing Solutions,“ rev. *33rd International Conference on Information Technology Interfaces*.

17] „CloudStack vs OpenStack: Comparing Open Source Cloud Infrastructure Management Solutions,“ [Online]. Available: https://openmetal.io/resources/blog/cloudstack-vs-openstack/#:~:text=1.%20All,CloudStack.

18] „loudStack vs. OpenStack Comparison – What you need to know before choosing a cloud management system,“ [Online]. Available: https://www.shapeblue.com/cloudstack-vs-openstack-

comparison/#:~:text=Image%3A%20OpenStack%20ArchitectureOpenStack%20is%20broken,components%20depending%20on%20your%20needs.

19] „Apache CloudStack. (n.d.). Architecture Overview,“ [Online]. Available: https://cloudstack.apache.org/.

20] „Proxmox VE. (n.d.). Proxmox VE Architecture,“ [Online]. Available: https://pve.proxmox.com/wiki/Main_Page.

21] „Proxmox vs OpenStack: Comparing Open Source Cloud Software,“ [Online]. Available: https://openmetal.io/resources/blog/proxmox-vs-openstack-comparing-open-source-cloud-software/#:~:text=1,provides%20a%20comprehensive%20suite%20of.

22] „ubernetes. (n.d.). Kubernetes Architecture.,“ [Online]. Available: https://kubernetes.io/docs/concepts/architecture/.

23] „OpenStack. (n.d.). Heat Orchestration Service.,“ [Online]. Available: https://docs.openstack.org/heat/.

24] „Terraform Registry. (n.d.). OpenStack Provider.,“ [Online]. Available: https://registry.terraform.io/providers/OpenStack/openstack/latest/docs.

25] „Ansible. (n.d.). OpenStack Modules,“ [Online]. Available: https://docs.ansible.com/ansible/latest/collections/openstack/cloud.

26] „ Apache CloudStack. (n.d.). API Documentation.,“ [Online]. Available: https://cloudstack.apache.org/api/index.html.

27] „Terraform Registry. (n.d.). CloudStack Provider.,“ [Online]. Available: https://registry.terraform.io/providers/cloudstack/cloudstack/latest/docs.

28] „Ansible. (n.d.). CloudStack Modules.,“ [Online]. Available: https://docs.ansible.com/ansible/latest/collections/community/general/.

29] „ Proxmox VE. (n.d.). Proxmox VE API Documentation,“ [Online]. Available: https://pve.proxmox.com/pve-docs/api-viewer/.

30] „GitHub. (n.d.). Telmate/terraform-provider-proxmox,“ [Online]. Available: https://github.com/Telmate/terraform-provider-proxmox.

31] „Ansible Galaxy. (n.d.). Proxmox modules/roles,“ [Online]. Available: https://galaxy.ansible.com/search?keywords=proxmox.

32] „Kubernetes. (n.d.). Concepts: Workloads,“ [Online]. Available: https://kubernetes.io/docs/concepts/workloads/.

33] „Kubernetes. (n.d.). Operators,“ [Online]. Available: https://kubernetes.io/docs/concepts/extend-kubernetes/operator/.

34] „erraform Registry. (n.d.). Kubernetes Provide,“ [Online]. Available: https://registry.terraform.io/providers/hashicorp/kubernetes/latest/docs.

35] „ Ansible. (n.d.). Kubernetes Modules. D,“ [Online]. Available: https://docs.ansible.com/ansible/latest/collections/kubernetes/core/.

36] „1000 Compute nodes resource scalability testing,“ [Online]. Available: https://docs.openstack.org/developer/performance-docs/test_results/1000_nodes/index.html#:~:text=We%20have%208%20instances%20of,13%20nodes.

37] „OpenStack. (n.d.). OpenStack Operations Guide - Scaling your Cloud,“ [Online]. Available: https://docs.openstack.org.

38] „Capacity planning and scaling,“ [Online]. Available: https://docs.openstack.org/operations-guide/ops-capacity-planning-scaling.html.

39] „Apache CloudStack. (n.d.). Architectural Overview,“ [Online]. Available: https://cloudstack.apache.org/.

[40] „Performance Evaluation of the CloudStack Private Cloud," rev. *DOI: 10.11591/closer.v2i6.5661*.

[41] „Proxmox VE. (n.d.). Proxmox VE Cluster," [Online]. Available: https://pve.proxmox.com/wiki/Cluster_Manager.

[42] „Clustering in Proxmox VE," [Online]. Available: https://hostman.com/tutorials/clustering-in-proxmox-ve/.

[43] „Proxmox forum," https://forum.proxmox.com/threads/proxmox-scalability-max-clusters-in-a-datacenter.122541/.

[44] „Kubernetes. (n.d.). Kubernetes Scale Targets," [Online]. Available: https://kubernetes.io/docs/setup/best-practices/cluster-large/.

[45] „Kubernetes. (n.d.). Autoscalin," [Online]. Available: https://kubernetes.io/docs/concepts/workloads/controllers/autoscaler/.

[46] „OpenStack Foundation. (n.d.). User Stories & Case Studies.," [Online]. Available: https://www.openstack.org/users/.

[47] „VMware vs OpenStack: Why OpenStack is the better VMware alternative in 2024," [Online]. Available: https://platform9.com/blog/openstack-vs-vmware/.

[48] „OpenStack. (n.d.). OpenStack Development Cycle and Testing," [Online]. Available: https://docs.openstack.org/dev-docs/.

[49] „Apache CloudStack. (n.d.). About CloudStack," [Online]. Available: https://cloudstack.apache.org/about.html.

[50] „Apache CloudStack. (n.d.). CloudStack Installation Guide - High Availability.," [Online]. Available: https://docs.cloudstack.apache.org/en/latest/installguide/ .

[51] „Apache CloudStack. (n.d.). User Stories.," [Online]. Available: https://cloudstack.apache.org/users/index.html.

[52] „Proxmox VE. (n.d.). About Proxmox VE," [Online]. Available: https://pve.proxmox.com/wiki/Main_Page.

[53] „Proxmox VE. (n.d.). High Availability Cluster," [Online]. Available: https://pve.proxmox.com/wiki/High_Availability_Cluster.

[54] „ubernetes. (n.d.). User Stories," [Online]. Available: https://kubernetes.io/case-studies/.

[55] „Kubernetes. (n.d.). High Availability Kubernetes Clusters," [Online]. Available: https://kubernetes.io/docs/setup/production-environment/tools/kubeadm/high-availability/.

[56] „oudFerro builds its clouds based on OpenStack - widely deployed cloud framework," [Online]. Available: https://cloudferro.com/cloud/openstack/#:~:text=Industry,APIs.

[57] „CloudStack Installation options," [Online]. Available: https://docs.cloudstack.apache.org/en/4.17.0.0/installguide/optional_installation.html#:~:text=AWS%20API%20User%20Setup%C2%B6.

[58] „CNCF Research Reveals How Cloud Native Technology is Reshaping Global Business and Innovation," [Online]. Available: https://www.cncf.io/announcements/2025/04/01/cncf-research-reveals-how-cloud-native-technology-is-reshaping-global-business-and-innovation/#:~:text=CNCF%20Research%20Reveals%20How%20Cloud,CI%2FCD%20adoption%20is%20fueling.

[59] „https://openmetal.io/resources/blog/cloudstack-vs-openstack/#:~:text=1.%20All,set%20up%20and%20deploy%20compared," [Online]. Available: https://openmetal.io/resources/blog/cloudstack-vs-openstack/#:~:text=1.%20All,set%20up%20and%20deploy%20compared.

60] „CloudStack vs OpenStack: How Do They Compare To Each Othe,“ [Online]. Available: https://openmetal.io/resources/blog/cloudstack-vs-openstack/#:~:text=However%2C%20this%20flexibility%20comes%20at,times%20and%20increased%20operational%20complexity.

61] „OpenStack Keystone Authentication Using LDAP,“ [Online]. Available: https://platform9.com/blog/keystone-authentication-ldap/.

62] „OpenStack MFA,“ [Online]. Available: https://docs.openstack.org/keystone/latest/user/multi-factor-authentication.html.

63] „Apache CloudStack Features,“ [Online]. Available: https://cloudstack.apache.org/features/#:~:text=Multi.

64] „Kubernetes vs OpenStack: which one to choose?,“ [Online]. Available: https://ubuntu.com/blog/kubernetes-vs-openstack#:~:text=OpenStack%20is%20a%20cloud%20platform,institutions%2C%20manufacturing%20companies%20and%20governments.