NAGIOS CORE SERVER ZA NADZOR IT SUSTAVA

Plećaš, Mladen

Undergraduate thesis / Završni rad

2020

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Algebra University College / Visoko učilište Algebra**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:225:081552

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-18**

Repository / Repozitorij:

Algebra Univerity College - Repository of Algebra Univerity College

VISOKO UČILIŠTE ALGEBRA

ZAVRŠNI RAD

Nagios Core server za nadzor IT sustava

Mladen Plećaš

Zagreb, veljača 2020.

„Pod punom odgovornošću pismeno potvrđujem da je ovo moj autorski rad čiji niti jedan dio nije nastao kopiranjem ili plagiranjem tuđeg sadržaja. Prilikom izrade rada koristio sam tuđe materijale navedene u popisu literature, ali nisam kopirao niti jedan njihov dio, osim citata za koje sam naveo autora i izvor, te ih jasno označio znakovima navodnika. U slučaju da se u bilo kojem trenutku dokaže suprotno, spreman sam snositi sve posljedice uključivo i poništenje javne isprave stečene dijelom i na temelju ovoga rada".

U Zagrebu, datum.

Predgovor

Ovom prilikom želim se zahvaliti svojem mentoru na stručnom vodstvu, uloženom vremenu i trudu. Svojim kolegama i profesorima koji su nesebično dijelili svoje znanje i iskustvo.

Najviše bi se zahvalio svojoj ženi i obitelji koji su mi je bili najveća podrška kroz cijeli period studiranja

Prilikom uvezivanja rada, Umjesto ove stranice ne zaboravite umetnuti original potvrde o prihvaćanju teme završnog rada kojeg ste preuzeli u studentskoj referadi

Sažetak

U završnom radu je opisan Nagios Core sustav otvorenog koda za nadzor mrežnih uređaja i servisa. U našem slučaju to su računalne učionice i potrebna poslužiteljska infrastruktura.

U prvom djelu rada opisuje se okolina u kojoj je potrebno implementirati sustav, te njezina kompleksnost, veličina i trenutne metode nadzora i kontrole.

Drugi dio obuhvaća dizajn i pripremu virtualne okoline, instalaciju pojedinih dijelova sustava, kao i sam Nagios Core poslužitelj.

Završni dio opisuje načine i metode testiranja, iniciranje problema i krajnje rezultate testiranja.

Summary

The final paper describes the Nagios Core open source for monitoring network devices and services. In our case, these are computer classrooms and required server infrastructure.

First part of the paper desribes the endvironment in which system needs to be implemented, its complexity, size and current monitoring and control methods.

The second part covers the design and preparation of the virtual enviroment, instalation of individual parts of system, as well as Nagios Core server.

Final section describes methods and the way we test them, problem initiation and test results

Ključne riječi: Nagios, okolina, Windows, poslužitelj.

Sadržaj

1. Uvod 3

2. Opis problema 4

2.1. Okolina 5

2.2. Korišteni OS 7

2.3. Metode i mjerenja 8

3. Dizajn rješenja 10

3.1. Dizajn i planiranje rješenja 10

3.2. Instalacija okoline 12

4. Implementacija rješenja 15

4.1. Instalacija poslužitelja 15

4.2. Instalacija dodataka 22

4.3. Instalacija agenata 24

4.4. Generiranje izvještaja 29

5. Testiranje 33

5.1. Metode testiranja 33

5.2. Provođenje testova 33

5.3. Rezultati 35

Zaključak 37

Popis kratica 38

Popis slika 39

Popis tablica 41

Popis kôdova 42

Literatura 43

1. Uvod

U današnje vrijeme u svakom poslovnom okruženju veliki dio infrastrukture čine informacijski sustavi. Što sa sobom nosi određenu količinu poslužitelja, mrežnih uređaja i računala. Svaki dobro dizajniran sustav sadrži i elemente nadzora opreme i način na koji će se ta infrastruktura koristiti. Kada se definiraju ti parametri, osmišljavamo rješenje koje bi trebalo obuhvatiti parametre kao što su:

- Implementacija – mogućnost implementacije na infrastrukturu koju posjedujemo
- Cijena – isplativost u obliku održavanja i cijene licence
- Funkcionalnost – pokriva što više elemenata nadzora
- Mogućnost prilagođavanja – može pratiti rast firme i njezinih potreba
- Kompatibilnost – pruža mogućnost prilagođavanja na nove tehnologije i načine rada

Kada smo pronašli sustav koji zadovoljava parametre koje smo naveli gore, moramo razmisliti o načinu na koji ćemo ga koristiti:

- Sustav bi trebao moći generirati zapise o ispravnosti rada infrastrukture koju nadziremo.
- Mogućnost komunikacije sa administratorima sustava putem tekstualnih poruka ili slanjem elektroničke pošte
- Ispis izvještaja po parametrima koje mu zadamo u svrhu raznih analiza
- Uslijed nekog kvara da obavijesti nadležnu osobu ili sam otkloni problem

Nagios Core se nametnuo kao jedno od takvih rješenja, te smo ga odlučili implementirati i testirati u našem produkcijskom okruženju.

2. Opis problema

Rad u okruženju u kojem se nalazi veliki broj računala svakodnevni je stres za veliki broj administratora. Nakon godina iskustva i dobre organizacije posla, probleme možete svesti na minimum, ali uvijek se može dogoditi nešto nepredviđeno. Korisnik konkretnog računala koje je u kvaru često nema razumijevanja jer očekuje da će dobiti kvalitetnu i profesionalnu uslugu. Razumljivo je da netko tko ne razumije kako su računala u učionicama postavljena ili je tek početnik u radu sa računalom može zbog kvara izgubiti koncentraciju, a što će uzrokovati otežano praćenje nastave.

Najčešći problemi sa kojima se susrećem svaki dan:

- Nedostatak diskovnog prostora na računalu koji može biti uzrokovan nesvjesnim radnjama korisnika. Dobar primjer je rad sa Adobe alatima, koji automatski imaju podešeno čuvanje materijala na C:\ sistemsku particiju. Većina alata iz te skupine generiraju datoteke koje zauzimaju veliki količinu prostora na disku, a korisnik je naučen da čuva datoteke na predodređenoj putanji. Problem se može manifestirati i na D:\ particiji prilikom rada sa virtualizacijskim alatima. Korisnik može više puta sačuvati stanje virtualne mašine i pritom će generirati dodatne datoteke koje jako brzo mogu popuniti disk.
- Loš rad ili opterećenost radne memorije. Prilikom rada na određenim alatima koji zahtijevaju veliku količinu radne memorije može doći do prestanka rada. U većini slučajeva prašina zna uzrokovati loš rad radne memorije koja onda direktno utječe na računala.
- Iskoristivost računala u učionicama. Ovo je bitna stavka jer u većini slučajeva korisnici ne gase računala što može generirati veliku potrošnju struje. Također ako se neko računalo nije ugasilo duži period može doći do neispravnog rada jer nije u potpunosti instaliralo zakrpe.
- Problemi se manifestiraju i na nekim poslužiteljima koji su bitni za ispravno funkcioniranje učionica, ali nisu neophodni za njihov ispravan rad. Konkretan primjer je WSUS poslužitelji koji je je zadužen za instalaciju zakrpa na učionička računala. Pošto se nalazi na udaljenim poslužiteljima u virtualizacijskoj okolini često se ne primijeti prestanak rada jer nemamo sustav upozorenja.

Nameće se i pitanje koliko se često kvarovi događaju? Iako unutar firme postoje načini da se kvarovi prate kroz prijave putem sustava elektroničke pošte, na taj način gubi se vrijeme na skupljanje informacija i konzultiranje sa kolegama. Ako je neko računalo više puta bilo u kvaru, a svaki puta ga je popravljao drugi tehničar ne može se definirati učestalost ponavljanje istog kvara.

2.1. Okolina

Za potrebe završnog rada koristimo infrastrukturu Algebra grupacije. Pošto se firma bavi informatičkom edukacijom, posjeduje veliku količinu računala, poslužitelja i mrežne opreme. Svakog dana veliki broj zaposlenika, studenata i polaznika koristi računalnu opremu u poslovnicama diljem Hrvatske. Trenutačne lokacije su u Zagrebu, Varaždinu, Osijeku, Puli Rijeci, Zadru, Šibeniku i Splitu. Sva računala su dodana u dvije domene, *algebra.local i ucione.local,* kako bi povećali razinu sigurnosti i upravljivost sustava.

Algebra.local domenu koriste stalno zaposlene osobe unutar Algebra grupacije u svim poslovnicama na razini Hrvatske. Mreža je fizički i logički odvojena od ucione.local mreže. Logika naziva računala je LOKACIJA-ODJEL-IME i računala dobivaju IP adrese preko DHCP-a. Jedan korisnik koristi uvijek isto računalo. Nadzor i upravljanje računalima zaposlenika vrše sistemski administratori i nemaju previše doticaja sa učionicama. Na *algebra.local* domenu nemaju pravo pristupa osobe koje nisu zaposlene u Algebra grupaciji.

Ucione.local je domena u kojoj se nalaze sva računala u učionicama, dvoranama i kabinetima. Da bi tolika količina računala bila upravljiva, od strane sistemskih tehničara, mora postojati razumljiv i logičan dogovor oko načina povezivanja poslovnica na centralni sustav, jedinstvena nomenklatura za nazive računala i strogo definirana mreža. Nazivi računala su najčešće definirana po principu POSLOVNICAUČIONICA-BROJRAČUNALA:

- ILICA - zbog velikog broja računala i količine tečajeva, programa obrazovanja i studijskih programa ova lokacija se razlikuje od ostalih. Pošto diplomski studiju predavanja idu na engleskom jeziku, morali smo lokalno napraviti neke izmjene i imena računala i dvorana prilagoditi stranim studentima.

  Primjer imena računala u učionici je CR05-04.ucione.local. CR je oznaka da je to učionica. Poslije toga je dodan broj učionice od 00 do 06 u prizemlju i od 11 do 17 na prvom katu. Broj računala koji se stavlja na kraju je definiran brojem 00 za predavača i

brojevima 01 do 30 (trenutačno je to najveća količina računala u nekoj učionici) za polaznička računala. Trenutačno brojimo oko 307 računala u 14 učionica i kabineta.

Dvorane koriste nešto drugačiju definiciju imena, npr. LR-MC-01. LR označava da je to dvorana. MC je jedinstveni skraćeni naziv za dvoranu Marie Curie koji je definiran od strane dekanata. Na kraj smo stavili broj 01 zbog mogućnosti dodavanja dodatnih predavačkih računala. Trenutačno brojimo 7 dvorana sa 7 računala.

Iznimke u ovoj nazivlju su dvorana/učionica Slavoljub Penkala sa oznakama računala LR-SP-00 do LR-SP-30 i kabinet/učionica Ivan Lupis CRLUP-00 do LRLUP-18.

- MAKSIMIRSKA – druga najveća poslovnica poslije Ilice. 6 učionica sa 14+1 računalo. Nazivi su definirani skraćenim nazivom poslovnice, brojem učionice i brojem računala, npr. MAX02-03. Poslovnica se nalazi u Zagrebu pa je direktno pod nadležnošću administratora koji se nalaze u Zagrebu
- VARAŽDIN - 2 učionice sa ukupnim 27 računala. Nazivi su definirani skraćenim nazivom poslovnice, brojem učionice i brojem računala, npr. VZ02-00. Poslovnica je pod nadležnošću vanjskog tehničara i administratora u Zagrebu.
- OSIJEK - 2 učionice sa ukupnim 28 računala. Nazivi su definirani skraćenim nazivom poslovnice, brojem učionice i brojem računala, npr. OS02-07. Poslovnica je pod nadležnošću vanjskog tehničara i administratora u Zagrebu.
- PULA - 2 učionice sa ukupnim 28 računala. Nazivi su definirani skraćenim nazivom poslovnice, brojem učionice i brojem računala, npr. PU02-00. Poslovnica je pod nadležnošću vanjskog tehničara i administratora u Zagrebu.
- ZADAR - 2 učionice sa ukupnim 26 računala. Nazivi su definirani skraćenim nazivom poslovnice, brojem učionice i brojem računala, npr. ZD01-11.
- ŠIBENIK - 1 učionice sa ukupnim 13 računala. Nazivi su definirani skraćenim nazivom poslovnice, brojem učionice i brojem računala, npr. SI02-00.
- SPLIT - 3 učionice sa ukupnim 43 računala. Nazivi su definirani skraćenim nazivom poslovnice, brojem učionice i brojem računala, npr. ST01-13.

Mrežna infrastruktura i raspodjela *IP* adresa je složena na sljedeći način:

- ILICA – računala imaju *IP* adrese iz A klase 10.10.0.0, ali imaju *subnet* iz klase C, 255.255.255.0. Logika dodjeljivanja adresa je 10.10.A.B, gdje je A broj učionice, a B broj računala. Dvorane imaju adrese oblika 10.10.C.D, C je *subnet* koji je najbliži toj dvorani, a D je adresa između .200 i .205.

Iako neke dvorane i učionice ne prate ovu logiku, zbog učestalih nadogradnji i mijenjanja naziva učionica i dvorana, računala imaju fiksne *IP* adrese pa se lako može pratiti raspodjela.

- MAKSIMIRSKA – računala imaju *IP* adrese iz C klase 192.168.0.0 i *subnet* iz klase C, 255.255.255.0. Logika dodjeljivanja adresa je 192.168.A.B, gdje je A broj učionice, a B broj računala.
- POSLOVNICE - računala imaju *IP* adrese iz B klase 172.19.0.0 i *subnet* iz klase C, 255.255.255.0. Logika dodjeljivanja adresa je 172.19.A.B, gdje je A broj učionice, a B broj računala.

Sve poslovnice su povezene putem SSL tunela, a što nam omogućava Fortigate uređaj koji spada u vatrozid nove generacija. Navedeno nam je iznimno bitno jer pomoću njega imamo pristup svim računalima u učionicama sa jednog centralnog mjesta.

2.2. Korišteni OS

Algebra grupacija, kao Microsoft partner, pretežno cijelu infrastrukturu ima na nekom Microsoft operativnom sustavu. Razlog tome je i što većina edukacije koju pruža vezana za neki od Microsoft alat ili rješenja.

Poslužitelji vezani za *ucione.local* domenu imaju instalirane Windows Sever operacijske sustava od verzije 2012 do 2019. Smješteni su u virtualizacijsku okolinu koja se nalazi na fizičkom poslužitelju. Iako postoje poslužitelji na Linux operacijskom sustavu, služe više kao testna okruženja.

Računalne učionice kao primarni operacijski sustav koriste Windows 10 Enterprise. Koristimo ga zbog:

- Napredne zaštite od modernih sigurnosnih ugroza
- Fleksibilnosti kod instalacije i nadogradnje
- Kvalitetne podrške
- Jednostavnog upravljanja aplikacijama i uređajima

Trenutačna verzija je 1903 na većini računala. Iako je izašla verzija 1909, nismo je implementirali zbog nekih problema koji su uočeni tokom testiranja.

Sva računala na primarnom operacijskom sustavu su pridružena domeni *ucione.local,* radi centraliziranog upravljanja organizacijskim jedinicama (učionice), korisnicima (računala) i razinama pristupa resursima na poslužiteljima i računalima.

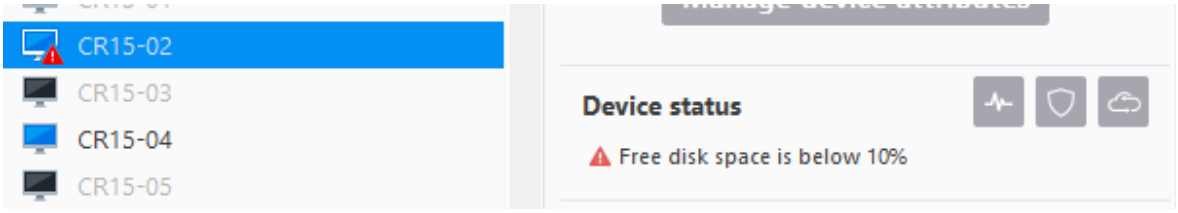
Većina računala na sebi ima postavljen i drugi operacijski sustav Windows 10 Professional kojeg koristimo za edukacije vezane za Android ili Cisco programe obrazovanja. Na računala se postavljaju u obliku *.vhd* ili *.vhdx* slika i ne dodajemo ih u domenu. Zbog specijalnih uvjeta koji moraju biti zadovoljeni, računala ne stavljamo u domenu.

Neke učionice u kojim a se odvija specijalistička edukacija vezana za Microsoft proizvode imaju implementiran Windows Server 2012, 2016 i 2019 verzije operacijskih sustava. Postavljaju se u obliku *.vhd* i *.vhdx* slika. Služe nam kao virtualne testne okoline.

2.3. Metode i mjerenja

Pošto unutar učionica nemamo informatički sustav za nadzor i mjerenje rada računala i poslužitelja, osmislili smo sustave provjere pomoću lista provjere. Na centralnom mjestu unutar Sharepoint sustava imamo kreirane interaktivne liste za sve učionice i dvorane. Na tjednoj bazi se učionice provjeravaju po točkama koje su navedene na listama. Provjerava se fizička ispravnost računala, zdravlje operacijskog sustava i stanje strujnih i mrežnih kabela. Ovakva metoda u nekim slučajevima nije efikasna jer postoji faktor čovjeka. Za razliku od računalnih sustava koji vrše nadzor, ljudi su skloniji pogreškama. Nekada je učionica nedostupna pa se mora raditi vikendima ili noću, a da bi to izbjegli provjere znaju trajati više dana zbog količine računala. U nekim periodima tokom godine nema dovoljno ljudstva i vremena da bi se provjerile sve učionice pa se otklanjaju samo najkritičniji problemi. Svejedno ova metoda provjere se pokazala iznimno učinkovitijom od strojne jer se provjeravaju i fizički elementi učionice. Kroz razne ankete koje provjeravaju zadovoljstvo korisnika mogu se dobiti mjerljivi podaci o efektivnost ovakve metode provjere.

Druga metoda je program Teamviewer. Alat koji pruža sigurno spajanje i upravljanje na računala na daljinu. Omogućava grupiranje i brzu pretragu računala po imenima. Instaliranjem agenata na računala omogućava se jednostavan pristup računalu u bilo kojoj poslovnici u bilo koje vrijeme. Osim što omogućava jednostavan pristup, podesili smo i nadzor računala. U ovom trenutku možemo nadzirati zapunjenost sistemske particije i da li je antivirusni program omogućen/instaliran. Notifikacije su vidljive samo prilikom spajanja na sustav.



Slika 2.1.Prikaz TeamViewer obavijesti

Prednosti su brzi pristup računalu, mogućnost pristupa bilo kojem profilu na računalu i notifikacija su vidljive prije samog spajanja na računalo. Negativnost je velika cijena licence i nemogućnost slanja obavijesti o stanju računala na adresu elektroničke pošte korisnika.

3. Dizajn rješenja

Zbog kompleksnosti sustava i velike količine računala dizajn smo bazirali u četiri točke

- Nadziremo poslužitelje i računala unutar učionica. Za poslužitelje ćemo nadzirati dostupnost samog poslužitelja, rad servisa i korištenje diskovnog prostora. Na računalima nadziremo diskovni prostor, radnu memoriju i procesore.
- Nadziremo ih kroz Nagios Core web sučelje, gdje su nam vidljiva stanja uređaja i servisa. Metode dodavanja računala unutar sučelja smo napravili kroz par alata.
- Obavijesti se šalju na adresu elektroničke pošte korisnika i na Nagios Log poslužitelj koji nam omogućava lakši pregled i bržu analizu.
- Izrada izvještaja o dostupnosti uređaja i servisa. Izvještaji se generiraju preko web sučelja na Nagios Core – u ili generiranje .csv datoteka.

3.1. Dizajn i planiranje rješenja

Najčešći problemi sa kojima se svakodnevno susrećemo u učionicama su nedostatak prostora na diskovima, problemi sa opterećenjem komponenti i dostupnosti nekih poslužitelja koji su neophodni za funkcioniranje sustava. Kao rješenje konfigurirao sam Nagios Core poslužitelj otvorenog koda za praćenje sustava i mreža. Iskoristili smo i sustav upozorenja koji nas obavještava ako se dogodi nešto loše ili dobro. Iako koristimo puno njegovih funkcionalnosti treba navesti koje su mu mogućnosti i koje ćemo u budućnosti implementirati:

- Nadziranje servisa na mreži
- Nadzor računalnih resursa (opterećenje procesora ili zapunjenost diska)
- Jednostavnost dodataka koje daje mogućnost izrade svojih vlastitih provjera
- Paralelno praćenje servisnih provjera
- Mogućnost definiranja računala koja su nedostupna i ona koja su ugašena
- Slanje notifikacija kada se problem dogodi ili riješi
- Automatsko rotiranje log datoteka

Prije nego smo počeli sa dizajnom rješenja morali smo objasniti dva najbitnije elementa po kojima kreiramo cijelo rješenje.

- *Host* je bilo koji virtualni ili fizički uređaj na mreži. Smjestili smo ih u ogovarajuće *host* grupe. Morali smo odvojili računala od poslužitelja s obzirom da nadziremo različite stvari.

- *Service* su funkcionalnosti, servisi na fizičkim ili virtualnim uređajima direktno vezanih za njih. Također smo ih morali odvojiti u servisne grupe.

Nakon što smo to definirali moramo objasniti i sustav obavijesti servisa koji se sastoji od 4 vrste stanja.: *OK, Warning, Critical* i *Unknown*.

Koristimo *Ok*, *Warning* i *Critical* jer time smanjujemo količinu podataka na koju moramo obraćati pozornost, skraćuje vrijeme reakcije i lakše analiziramo problem. Obično kada je stanje *Unknown* pogriješili smo negdje u konfiguraciji. Što smo preciznije definirali neke vrijednosti unutar upozorenja to smo brže mogli reagirati. Dobar primjer je količina slobodnog prostora na C:\ ili D:\ *particiji* diska računala. Postavili smo *Warning* stanja na 80% slobodnog prostora i *Critical* stanje na 90% zapunjenost. Ako se ovakvo stanje pojavilo na više računala, možemo pretpostaviti da diskovni prostor koji smo odredili po particijama nije adekvatan za nastavu ili se dogodila neka greška te se pune podacima koji se ne bi trebali nalaziti na njima.

Soft and Hard state - Nagios omogućava precizno obavještavanje i šalje je samo u slučaju da je stanje servisa ili uređaja u *Hard state-u* . Ovo nam je iznimno bitno kod dizajna rješenja jer o tome ovisi točnost obavijesti. Funkcionira tako da se provjera odvija u tri faze. Prve dvije provjere su *Soft state*, a treća provjera je *Hard state*. Objasniti ćemo na primjeru resetiranja računala. U prvoj provjeri će sustav registrirati da su servisi u *Unknown* stanju jer ih računalo gasi i nisu dostupni. Kod druge provjere računalo se podiže, ali još uvijek nije diglo sve servise i pokazuje da su servisi *Unknown* stanju. Kod treće provjere računalo je došlo do *log on* ekrana i svi servisi su se digli i registrira se OK stanje. Ne šalje nikakvu obavijest stanja. Da se računalo slučajno nije podignulo *Hard* stanje bio bi *Down* i obavijest bi bila poslana na mail administratora sustava.[1]

Pošto Algebra raspolaže sa velikom količinom uređaja i njihovih servisa bitno je i da smo dobro definirali tko će primati kakve obavijesti i kada. Osoba koja je zadužena za poslužiteljsku infrastrukturu je dobila obavijest vezanu za poslužitelje, kao što je dostupno DNS-a i dostupnost samog poslužitelja. Osobe zadužene za učionice su dobili obavijest o iskorištenosti diskovnog prostora, procesorske snage ili radne memorije. Svaka osoba je odredila koji parametri su bitni za njegov dio poslovanja. U našem slučaju sve obavijesti su dolazile na mail adresu mladen.plecas@algebra.hr.

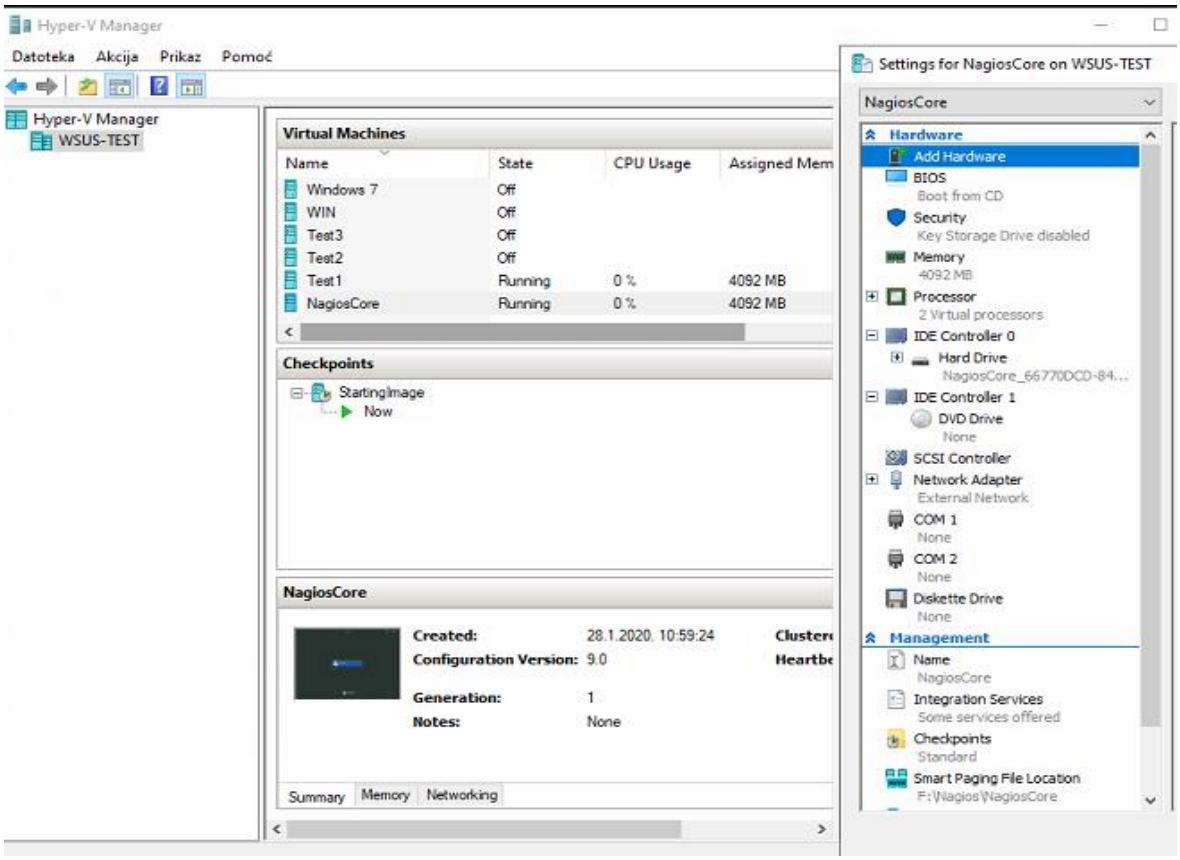
[1] Wojcieh Kocjan, Piotr Beltowski. Learning Nagios. Packt Publishing Ltd, 2016. Str 14.

Još jedan bitan element za dizajn i planiranje rješenja su konfiguracijske datoteke i pojmovi koji se nalaze u njima:

- *Commands* – definirane naredbe po kojima Nagios izvršava određene provjere. Nalazi se na putanji */usr/local/nagios/etc/objects* i nalaze se u *command.cfg* konfiguracijskoj datoteci.
- *Time periods* – vrijeme u kojem će se izvršavati provjere ili slati obavijesti putem tekstualne poruke ili porukama elektroničke pošte. Nalazi se na putanji */usr/local/nagios/etc/objects* i nalaze se u *timeperiods.cfg* konfiguracijskoj datoteci.
- *Services i services groups* – servisi koje pratimo na uređajima, kao što je diskovni prostor ili iskorištenost CPU-a. više servisa može biti u jednoj servisnoj grupi.
- *Host i Host groups* – definirani uređaji i njihove grupe. Jedan uređaj može biti u više grupa.
- *Contacts i contacts groups* - kontakti ljudi koji će biti obavješteni, na koji način i kada. Kontakti mogu biti grupirani i svaki zasebno može biti član više grupa
- *Notifications* – obavijesti. Obavijest može biti prikazana na web sučelju Nagios-a ili poslana administratoru sustava.
- *Escalations* – eskalacija obavijesti. Stanja u kojem se nalazi uređaj ili servis.

3.2. Instalacija okoline

Za potrebe završnog rada koristio sam računalo sa Windows 10 operacijskim sustavom, procesorom Core I5, 16 GB RAM-a i *Solid state* diskom od 500 GB. Računalo se nalazi u domeni *ucione.local* i ima pristup svim učionicama u poslovnicama. Računalo inače koristim za izradu i testiranje virtualnih mašina potrebnih za nastavu na Visokoj Školi Algebra. Obzirom da je na fizičkoj mašini podignut Windows operacijski sustav, a Nagios Core se može instalirati samo na Linux operacijskim sustavima, morao sam napraviti *virtualnu* mašinu. Za te potrebe sam koristio *Hyper-V* virtualizacijski alat. *Hyper-V* sam izabrao jer mi nije bila potrebna licenca i alat je sa kojem već godinama radim. Kod kreiranja virtualne mašine izabrao sam ime Nagios Core i odredio sam parametre: 2 virtualna procesora, 4 Gb RAM-a i diskovnim prostorom od 50 GB.

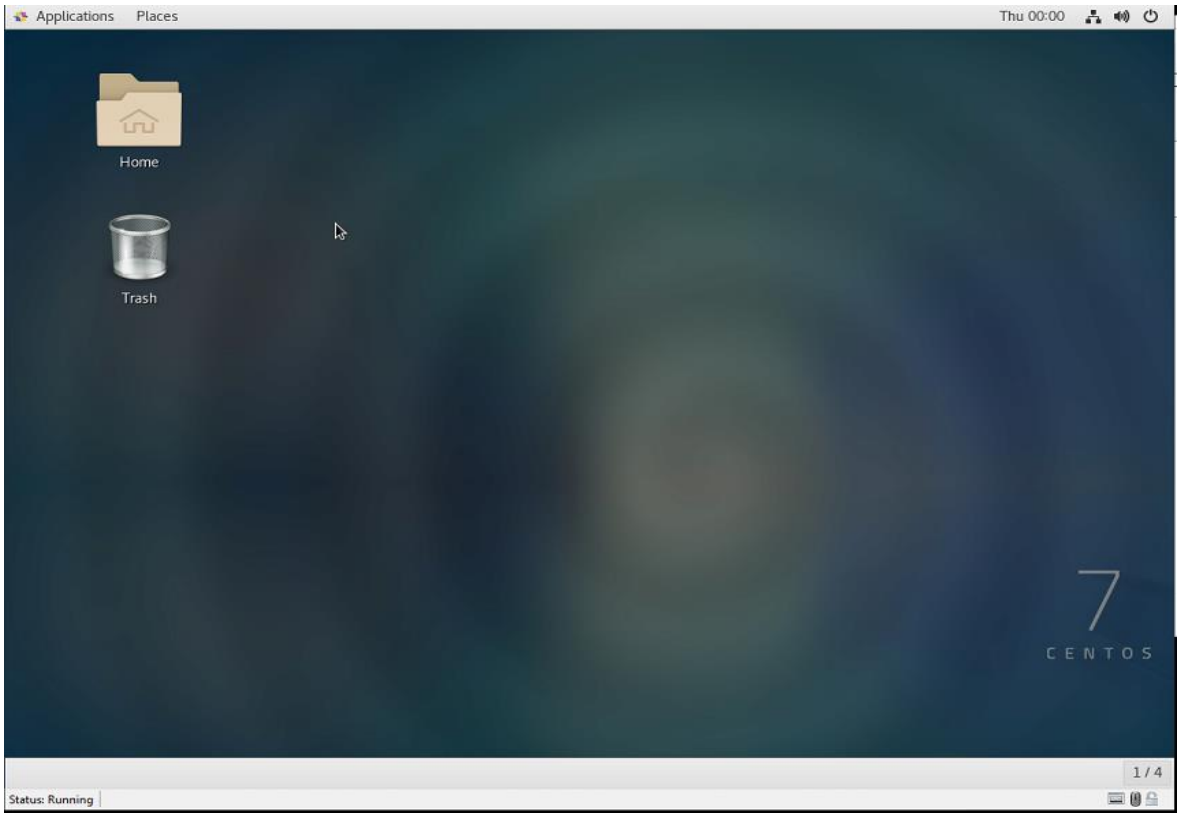


Slika 3.1 Podešavanje Hyper-V okruženja

Na virtualnu mašinu sam instalirao *CentOS* 7 (*The Commmunity ENTeprise Operating System*) operacijski sustav. Mogli sam instalirati bilo koji operacijski sustav otvorenog kdoa, kao što su *Ubuntu*, *Fedora*, *Ubuntu* i mnogi drugi.

OpenSource operacijski sustavi su potpunosti besplatni i mogu se kao takvi koristiti u poslovne i privatne svrhe. Kopije se mogu distribuirati potpuno besplatno, kao i bilo koja modificirana verzija.

Nakon što smo napravio virtualnu mašinu dodali smo *ISO* slika *CentOS-7-x85_64-Everithing -1804.iso* i pokrenuli instalaciju. Izabrali smo instalaciju sa *GNOM desktop-om* i *Security Tools* grupom. Tokom instalacije smo napravili novu šifru za korisnika root i stvorili novog korisnika Mladen bez privilegiranih ovlasti. Nakon završetka instalacije podesili smo mrežne postavke, ime i vrijeme. Mrežu smo postavili sa *IP* adresom 10.10.7.94 i ostalim postavkama koje inače stavljamo u učionice. Time smo omogućili pristup ostalim uređajima u učioničkoj mreži. Vrlo bitna stavka je i postavljanje točnog vremena na poslužitelja da bi obavijesti bile u stvarnom vremenu.



Slika 3.2 CentOS operacijski sustav

4. Implementacija rješenja

Implementacija Nagios Core sustava se može napraviti na više načina. Jedan od najjednostavnijih rješenja bila je instalacija preko skripti koje se mogu pronaći unutar raznih repozitorija na internetu ili u knjigama. Mana ovog načina instalacije je što su direktoriji i konfiguracijske datoteke prilagođene korisnicima koji su ih kreirali. U svrhu upoznavanja sa sustavom napravili smo dvije instalacije i na kraju odustali jer su verzije sustava zastarjele. Mi smo odabrali instalaciju iz izvora jer je svaki korak instalacije detaljno objašnjen i popraćen sa velikom količinom dokumentacije.

4.1. Instalacija poslužitelja

Prije nego smo počeli sa instalacijom Nagios Core -a morali sam podesiti *SELinux (Security-Enhanced Linux)* u *permissive* stanje jer bi neke postavke mogle smetati kod instalacije. SELinux je sigurnosna arhitektura koja omogućava administratorima kontrolu nad time tko može pristupiti sustavu. *Permissive* stanje će nam ispisivati upozorenja, ali neće blokirati određene dijelove sustava.

Instalacija najnovije verzije NagiosCore 4.4.5 se izvodi u sljedećim koracima:

- Prvo moramo skinuti par paketa koji su preduvjet za daljnje korake

```
yum install –y gcc glibc glibc-common wget unzip httpd php gd gd-devel perl postfix
```

- Instalacija se skida sa github.com repozitorija

```
cd /tmp

wget –O nagioscore.tar.gz httpd://github.com/NagiosEnterprises/nagioscore

/archive/nagios-4.4.5.tar.gz

tar xzf nagioscore.tar.gz
```

- Kompajliranje

```
cd /tmp/nagioscore-nagios-4.4.5/

./configure

make all
```

- Stvorimo *nagios* grupu i korisnika i dodamo *apache* korisnika u *nagios* grupu

```
make install-groups-users
```

```
usermod –a –G nagios nagios

usermod –a –G nagios apache
```

- Instaliramo binarne datoteke, CGIs i HTML datoteke:

```
make install
```

- Instaliranje servisa, konfiguriramo da se pokrenu za pokretanje mašine i pokretanje *Apache httpd servisa* .

```
make install-deamoninit

systemctl enable httpd.service
```

- Instaliranje i konfiguriranje vanjske upravljačke datoteke

```
make install-commandmode
```

- Instaliranje konfiguracijske datoteke koja je uzorak jer je Nagios treba da bi se pokrenuo

```
make install-config
```

- Instaliranje Apache web poslužitelja

```
make install-webconf
```

- Podešavanje vatrozida i otvaranje porta 80

```
firewall-cmd –zone=public –add-port=80/tcp

firewall-cmd –z[2]one=public –add-port=80/tcp –permanent
```

- Stvaranje lozinke za spajanje na Nagios:

```
htpasswd –c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

- Pokretanje Nagios Core-a

```
systemctl start nagios.service
```

Nakon uspješne instalacije spajamo se na web sučelje preko adrese http://10.10.7.94/nagios. Prije spajanja na server potrebno je bilo resetirati Apache web poslužitelj naredbom *systemctl restart httpd.service.*

[2] https://support.nagios.com/kb/article/nagios-core-installing-nagios-core-from-source-96.html

Slika 4.1 Nagios Core web sučelje

Nakon uspješne instalacije morali smo se upoznati sa strukturom direktorija i datoteka koje sadrže sve elemente neophodne za rad sustava.

Putanja	Opis
/usr/local/nagios/etc	Konfiguracijski direktorij
/sur/local/nagios/etc//nagios.cfg	Glavna konfiguracijska datoteka – sadrži konfiguracijske datoteke kao što su *log_file, cfg_file* i *cfg_dir*
/usr/local/nagios/var	Direktorij koji sadrži lokalni status Nagiosa
/usr/local/nagios/var/archives	Povijest informacija vezanih za Nagios. Veličina zna biti varijabilna pa je dobra praksa staviti ga na poseban disk
/usr/local/nagios/var /status.dat	Datoteka koja sadrži trenutni status Nagiosa
/usr/local/nagios/var /rw/nagios.cmd	Razdjeljak za pisanje naredbi u Nagios – u
/usr/local/nagios/share	Web korisničko sučelje koje treba poslužiti preko web servera, npr. *https://localhost/nagios*
/usr/local/nagios/sbin	CGI skripte koje bi trebale biti poslužene preko web pretraživača, npr. *https://localhost/nagios/cgi-bin*
/usr/local/nagios/libexec	Putanja do Nagios dodataka koje koristimo za izvršavanje naredbi

Tablica 4.1 Struktura direktorija i datoteka unutar Nagios Core-a

Nagios.cfg – glavna konfiguracijska datoteka sadrži jednostavnu sintaksu. Linija počinje sa # znakom koji je označava komentar i linije u formatu *<neki parametar>=<neka vrijednost>*, *npr. Log_file=/usr/local/nagios/var/nagios.log*. U nekim slučajevima se vrijednost može ponavljati. Definicije koje nam pomažu da si se log file finije podesio:

Parametri	Opis
log_file	Određuje koja će se zapis datoteka koristiti: *[localstatedir]/nagios.log*
cfg_file	Određuje koja će se konfiguracijska datoteka koristiti za definiciju objekta
cfg_dir	Određuje konfiguracijski direktorij u kojem će biti sve datoteke koje se trebaju koristiti za definiciju objekta
resource_file	Datoteka za skladištenje dodatnih makro definicija: *[sysconfdir]/resource.cfg*
temp_file	Putanja do privremene datoteke koja se koristi za privremene podatke: *[localstatedir]/nagios.tmp*
lock_file	Putanja do datoteka koja se koristi za sinkronizaciju: *[localstatedir]/nagios.lock*
temp_path	Putanja gdje Nagios može stvarati privremene datoteke: */tmp*
status_file	Putanja do datoteke koja sprema trenutačno stanje svih uređaja i servisa: *[localstatedir]/status.dat*
status_update_interval	Određuje koliko često se *status_file* treba ažurirati
nagios_user	Korisnik koji pokreće pozadinske procese
nagios_group	Grupa koja pokreće pozadinske procese
command_file	Određuje putanju do vanjskih komandnih linija kojim se koriste drugi procesi za kontrolu Nagios pozadinskih procesa: *[localstatedir]/rw/nagios.cmd*
use_syslog	Određuje da Nagios zapisuje zapise u *syslog* isto kao i u Nagios *log* datoteku
state_retention_file	Putanja do datoteke koja skladišti informaciju stanja diljem gašenja: *[localstatedir]/retention.dat*
retention_update_interval	Koliko često datoteka zadržavanja treba biti ažurirana
service_check_timeout	Za koliko sekundi provjera servisa treba pretpostaviti da nije uspjela
host_check_timeout	Za koliko sekundi provjera uređaja treba pretpostaviti da nije uspjela
event_handler_timeout	Za koliko sekundi rješavatelj događaja treba biti ugašen

notification_timeout	Za koliko sekundi bi trebalo pretpostaviti da je propao pokušaj obavijesti
enable_environment_macros	Da li bi Nagios trebao proslijediti sve makro naredbe na dodatke kao varijable okoline
interval_lenght	Određuje koliko je je sekundi jedinični interval: zadano je 60 sekundi

Tablica 4.2 Definicije za podešavanje Log datoteke

Nakon što smo objasnili neke bitne direktorije, datoteke i elemente pomoću kojih Nagios funkcionira počinjemo definirati uređaje, servise, kontakte, komande i vremenske periode.

Sve datoteke nam se nalaze u putanji */usr/local/nagios/etc/objects*

```
[mladen@nagioscore ~]$ cd /usr/local/nagios/etc/objects/
[mladen@nagioscore objects]$ ls
@              CR17-20.cfg         localhost.cfg         templates.cfg
commands.cfg   CR17-21.cfg         nagioslog.cfg         timeperiods.cfg
contacts.cfg   CR17-22.cfg         nagioslogserver.cfg   ucionice.cfg
CR05-02.cfg    hostgroups.cfg      printer.cfg           windows.cfg
CR05-04.cfg    hostoviucione.cfg   servicegroups.cfg     wsus-os.cfg
CR05-05.cfg    ili-nastava.cfg     switch.cfg
```

Slika 4.2 Direktor u kojem se nalaze konfiguracijske datoteke

Kao podlogu smo iskoristili *windows.cfg* datoteku s obzirom da ćemo za mašine koje imaju Windows operativni sustav koristiti NSClient++ agenta koji koristi *check_nt* način spajanja na Nagios Core.

```
define host {
    use                             windows-server
    host_name                       CR05-02.ucione.local
    alias                           ucionice
    address                         10.10.4.2
}
# nadzor CPU-a
define service {
    use                             generic-service
    host_name                       CR05-02.ucione.local
    service_description             CPU Load
    check_command                   check_nt!CPULOAD!-l
5,80,90
}
# nadzor RAM-a
define service {
    use                             generic-service
    host_name                       CR05-02.ucione.local
    service_description             Memory Usage
```

```
    check_command                   check_nt!MEMUSE!-w 80 -c
90
}
# nadzor C particije
define service {
    use                             generic-service
    host_name                       CR05-02.ucione.local
    service_description             C:\ Drive Space
    check_command                   check_nt!USEDDISKSPACE!-l
c -w 80 -c 90
}
# nadzor D particije
define service {
    use                             generic-service
    host_name                       CR05-02.ucione.local
    service_description             D:\ Drive Space
    check_command                   check_nt!USEDDISKSPACE!-l
d -w 80 -c 90}
```

Kod 4.1 Konfiguracija host datoteke

Logika je sljedeća. Prvi dio sa nazivom *define host*

- *use* *windows-server* – koriste se unaprijed definirane postavke iz *templates.cfg* datoteke pod nazivom *define host {name windows-server}*
- *host_name* *CR05-02.ucione.local – FQDN (Fully qualified domain name)* računala koje se dodaje u Nagios
- *alias* *ucionice* – ovo može biti bilo kakav naziv koji će nas asocirati na funkciju uređaja
- *address* *10.10.4.2* – IP adresa računala koje dodajemo

Druge stavke unutar naziva *define service{*

- *use* *generic-service* - koriste se unaprijed definirane postavke iz *templates.cfg* datoteke pod nazivom *define host {name generic-service}*
- *host_name* *CR05-02.ucione.local – FQDN (Fully qualified domain name)* računala koje se dodaje u Nagios
- *service_description* *CPU Load* – naziv servisa
- *check_command* *check_nt!CPULOAD!-l 5,80,90* – naredba je definirana ovdje

Za dodavanje više korisnika ovo može biti mukotrpan posao pa smo proces pojednostavili izradom konfiguracijske datoteke koju možemo kopirati više puta za određenog korisnika. Naredbom cp možemo na brzinu stvoriti puno .cfg datoteka. Izmjene koje moramo napraviti unutar datoteke su naziv računala i IP adresu. Kod definicije servisa moramo zamijeniti naziv računala i to možemo napraviti jednostavnom naredbom*: 1,$ -s/CR05-02.ucione.local/CR05-04.ucione.local/g*. Da bi Nagios bio svjestan novonastale datoteke moramo ga dodati u *nagios.cfg* datoteku koja se nalazi u putanji */usr/local/nagios/etc.*

```
#Definitions for monitoring a Windows machine
cfg_file=/usr/local/nagios/etc/objects/CR17-20.cfg
cfg_file=/usr/local/nagios/etc/objects/CR17-21.cfg
cfg_file=/usr/local/nagios/etc/objects/CR17-22.cfg
cfg_file=/usr/local/nagios/etc/objects/CR05-02.cfg
cfg_file=/usr/local/nagios/etc/objects/CR05-04.cfg
cfg_file=/usr/local/nagios/etc/objects/CR05-05.cfg
```

Kod 4.2 *Nagios.cfg* datoteka

Bitno je da se putanje slažu po redu i bez pravopisnih grešaka. Kako bi što jednostavnije ubacili veliki broj putanja do konfiguracijskih datoteka uređaja, radili smo kopiranje postojećih putanja. Nakon kopiranja je bitno izmijeniti zadnji dio koji se odnosi na ime konfiguracijske datoteke. Ovim načinom smanjujemo mogućnost pravopisne greške. Ako ne namjeravamo koristiti tu datoteku možemo je jednostavno komentirati pomoću # znaka na početku reda i *nagios.cfg* ga neće primjenjivati.

Instalirali smo i Nagios Log poslužitelj koji će nam biti od velike pomoći za sortiranje podataka koje Nagios Core generira. Preporuka je da se instalira na čistu mašinu pa smo napravili još jednu CentOS virtualnu mašinu kao na slici 3.2. Instalacija je jednostavna i izvršava se preko skripte:

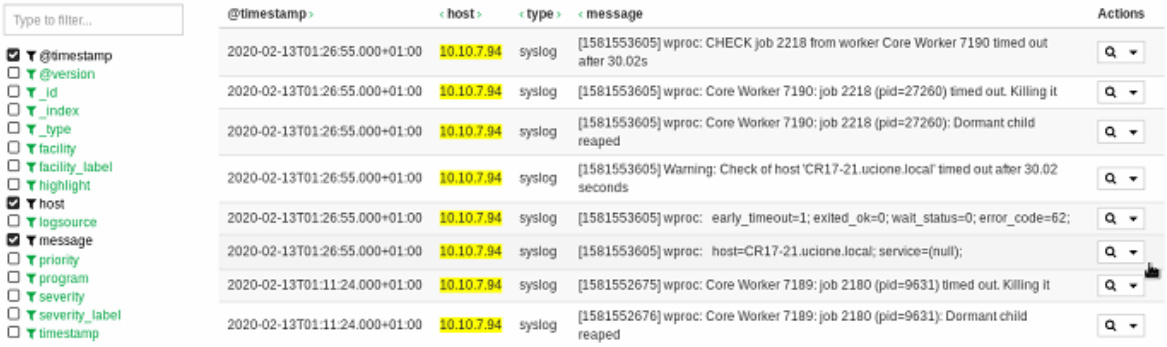
```
curl https://assets.nagios.com/downloads/nagios-log-server/install.sh | sh
```

Nakon što se skripta izvršila možemo se logirati na web sučelje preko adrese http://10.10.7.95/nagioslogserver. Na prvom ekranu smo odabrali opciju *Install*, nakon čega se pokazuje ekran sa odabirom licence i podešavanje administratorskog računa. Log

poslužitelj je besplatan dok god količina podataka ne prelazi 500MB po danu i ne kreiramo klastere. Kada smo podesili račun, možemo se prijaviti na poslužitelj.

Nagios Log i Core poslužitelj smo povezali u web sučelju Log poslužitelja. Otišli smo pod opciju *Global Config*, dodali novi filter kojeg smo nazvali Nagios Core i u njega kopirali filter koji se nalazi na repozitoriju githuba: https://github.com/T-M-D/NLS-Collection/blob/master/Filters/Nagios_Core.txt.[3]

Na početnoj stranici Log poslužitelja pod *Uniq Hosts/Reposrt* izabrali smo IP adresu našeg Core server i dobili smo filter događaja



Slika 4.3 Prikaz filtera događaja

Koristiti ćemo ga za detaljnije analize logova jer nudi mogućnost pristupa svakom zapisu pojedinačno, prikazuje detaljan opis problema i ima mogućnost pretraživanja interneta putem Google-a, Bing-a ili StackOwerflow-a.

4.2. Instalacija dodataka

Dodaci su nužni za rad Nagios Core poslužitelja i prije nego smo krenuli sa njihovom instalacijom morali smo ispuniti par preduvjeta:

```
yum install -y gcc glibc glibc-common make gettext automake autoconf wget openssl-devel net-snmp net-snmp-utils epel-release
```

```
yum install -y perl-Net-SNMP
```

Nakon izvršenih predradnji pokrenuli smo skidanje i instalaciju dodataka :

```
cd /tmp
```

[3] https://assets.nagios.com/downloads/nagios-log-server/docs/Sending-Nagios-Core-Logs-To-Nagios-Log-Server.pdf

```
wget -no-check-certificate -O nagios-plugins.tar.gz https://github.com/nagios-plugins/nagios-plugins/archive/release-2.2.1.tar.gz

tar zxf nagios-plugins.tar.gz

cd /tmp/nagios-plugins-release-2.2.1/

./tools/setup

./configure

make

make install
```

Svi dodaci se mogu locirati na putanji */usr/local/nagios/libexec*[4]

Da bi mogli koristiti dodatak moramo imati njegovu definiciju koja se upisuje u *commands.cfg*. Puno definicija dobijemo instalacijom Nagios Core-a i dodataka, ali za server ili-nastava-01.ucione.local smo dodali novu definiciju da bi vidjeli je li DNS servis funkcionalan:

```
Define command {
      Command_name      check_dns
      Command_line      $USER1$/check_dns -H 10.10.253.253 -s $HOSTADDRESS$
}
```

Druga definicija koja nam je potrebna za spajanje NSClient++ agenta na Windows mašinama sa našim služiteljem izgleda ovako i unaprijed je definirana. Dodali smo samo šifru koju smo definirali u nsclient.ini datoteci –s 12345:

```
Define command {
      command_name      check_nt
      comman_line       $USER1$/check_nt -H $HOSTADDRESS$ -p 12489 -s 12345 -v $ARG1$ $ARG2$}
```

Još jedan bitan dodatak je i sustav obavještavanja putem elektroničke pošte. Prvotna ideja bila je instalacija *Postfix SMTP* usmjerivača preko office365 i gmail servisa elektroničke pošte. Instalacija nije komplicirana, ali je generirala niz grešaka vezanih za sigurnosne politike kojima su zaštićeni ti servisi. Na kraju smo pronašli rješenje u obliku

[4] https://support.nagios.com/kb/article/nagios-core-installing-nagios-core-from-source-96.html#CentOS

sendmail.postfix. Doslovno je potrebno instalirati *Postfix* i u konfiguracijsku datoteku *commands.cfg* dodati */usr/sbin/sendmail.postfix*.

```
define command {

    command_name    notify-host-by-email
    command_line    /usr/bin/printf "%b" "***** Nagios host problem *****\n\nNotification Type: $NOTIFICATIONTYPE$\nHost: $HOS
TNAME$\nState: $HOSTSTATE$\nAddress: $HOSTADDRESS$\nInfo: $HOSTOUTPUT$\n\nDate/Time: $LONGDATETIME$\n" | /usr/sbin/sendmail.po
stfix  "** $NOTIFICATIONTYPE$ Host Alert: $HOSTNAME$ is $HOSTSTATE$ **" $CONTACTEMAIL$
}



define command {

    command_name    notify-service-by-email
    command_line    /usr/bin/printf "%b" "***** Nagios servisni problem*****\n\nNotification Type: $NOTIFICATIONTYPE$\n\nServi
ce: $SERVICEDESC$\nHost: $HOSTALIAS$\nAddress: $HOSTADDRESS$\nState: $SERVICESTATE$\n\nDate/Time: $LONGDATETIME$\n\nAdditional
 Info:\n\n$SERVICEOUTPUT$\n" | /usr/sbin/sendmail.postfix "** $NOTIFICATIONTYPE$ Service Alert: $HOSTALIAS$/$SERVICEDESC$ is $
SERVICESTATE$ **" $CONTACTEMAIL$
}
```

Slika 4.4 Podešavanje naredbe za slanje elektroničke pošte u *commands.cfg* datoteci

4.3. Instalacija agenata

Da bi mogli planirati instalaciju agenata vodili smo brigu o brzini instalacije, mogućnostima konfiguriranja i distribucije na veliki broj računala i o tome je li agent u pasivnom ili aktivnom načinu nadziranja sustava.

Aktivni način provjere se zasniva na sljedećem principu. Nagios pozadinski procesi iniciraju provjeru stanja uređaja ili servisa, aktivirat će dodatak koji je zadužen za tu provjeru i poslati će informaciju o tome što treba biti provjereno. Dodatak će tada provjeriti stanje servisa ili uređaja i poslati će rezultate Nagios pozadinskim procesima. Nagios će obraditi rezultate i poduzeti će odgovarajuće radnje. Aktivni pregled se može izvršavati po nekim zadanim postavkama koje su definirane sa *check_interval* i *retry_interval* opcijama na uređaju ili na zahtjev. Ako je servis u HARD stanju biti će aktivno pregledan po *check_interval* postavkama, a ako je u SOFT stanju pregledavati će se po *retry_interval* postavkama[5].

Pasivni način provjere je iniciran i izvršen od strane vanjske aplikacije i rezultati se onda prosljeđuju Nagios-u na obradu. Korisni su za servise koji su asinkroni po prirodi i ne mogu se efektivno nadzirati povlačeći stanja na redovnoj zadanoj bazi. Također su efektivni za uređaje koji se nalaze iza vatrozida i ne mogu se aktivno pratiti[6].

Za nadzor Windows 10 ili Windows Server operativne sustave, koristimo NSClient++ agenta. Iako ima opcije za pasivni nadzor, mi ga koristimo za aktivni način rada. Nije

[5] https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/3/en/activechecks.html
[6] https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/3/en/passivechecks.html

napravljen od s Nagios tima, ali je jednostavan za korištenje, podržava aktivni i pasivni nadzor i može se jednostavno distribuirati. Iako ima jako puno mogućnosti koristiti ćemo samo nadzor računala pomoću *check_interval (check_nt)*. Instalacija je sama po sebi jako jednostavna:

- sa stranice *https://nsclient.org/download/* smo skinuli najnoviju verziju agenta NSCP-0-5-2-35-x64.msi. Smjestili smo ga na centralni repozitorij koji je dostupan iz svih učionica za korisnika sa administratorskim ovlastima.
- dvostrukim klikom pokrenemo instalaciju i pratimo jednostavne korake
- na ekranu *Select Monitoring Tool* izabiremo *Generic* opciju
- Ekran *Choos Setup type* nam nudi opciju *Typical, Custom i Complete*. Pošto nam *Typical* instalacija nudi sve potrebne opcije za izvršavanje nadzora, druge dvije opcije ignoriramo.
- Pritiskom na opciju *Next* dolazimo do najvažnijeg ekrana *NSClient++ Configuration.* U polje *Allowed hosts* unesemo ostavimo opciju 127.0.0.1 i dodamo adresu NagiosCore-a odvojenu zarezom. *Password* polje ispunimo sa loznikom po želji i označimo sva polja sa kvačicama. Nama je najbitnija opcija *Enable nsclient server (check_nt)* jer omogućava komunikaciju klijenata i Nagios poslužitelja.
- Bitno je naglasiti da ako prilikom instalacije napravite greške u koracima, sve se može ispraviti.[7]

Nakon uspješne instalacije klijenta prikaz računala unutar web sučelja Nagiosa sustava će bacati grešku:

PU02-05.ucione.local	C:\ Drive Space	CRITICAL	02-12-2020 09:05:35	0d 0h 26m 12s	3/3	CRITICAL - Socket timeout
	CPU Load	CRITICAL	02-12-2020 09:07:25	0d 0h 24m 22s	3/3	CRITICAL - Socket timeout
	D:\ Drive Space	CRITICAL	02-12-2020 09:09:15	0d 0h 22m 32s	3/3	CRITICAL - Socket timeout
	Memory Usage	CRITICAL	02-12-2020 09:11:06	0d 0h 20m 41s	3/3	CRITICAL - Socket timeout

Slika 4.5 Neuspješno spajanje Nagios-a i uređaja pomoću NSClinet++ agenta

Da bi uspješno dovršili spajanje odlazimo u servise klijentskog računala i da pronađemo servis *NSClient++ Monitoring Agent.* Desnim klikom ulazimo pod Svojstva na karticu *Log On* i pod *Local System account* stavimo kvačicu na *Allow service to interact with desktop*. *Startup type* podesiti na *Automatic* da bi se servis dignuo svaki put prilikom paljenja računala i resetiramo ga.

[7] https://www.youtube.com/watch?v=zMBNdPBAMyA

Agent se instalirao na putanji c:\Programske datoteke\NSClient++. Unutar direktorija se nalazi datoteka *nsclient.ini* koja sadrži skriptu za izvršavanje nadzora poslužitelja i računala, dozvoljene IP adrese (adresa *NagiosCore* servera) i zaporku koju koristi *check_nt* za komunikaciju sa poslužiteljem. Da bi se ispravno izvršavala, morali smo napraviti neke izmjene. Sve opcije smo prebacili iz *disable* u *enable* ili *true* (1) i dodali smo dvije stavke koje se inače ne nalaze unutar datoteke:

```
; PERFORMANCE DATA - Send performace data back to Nagios (set this to 0 to
remove all performace data)

performace data = 1


[/settings/log]

file name =nsclient.log

level =debug
```

Prilikom dodavanja novih računala, iako su bila vidljiva na poslužitelju, NagiosCore poslužitelj nije mogao očitati stanje računala. Dodavanjem stavke *[settings/log]* u *nsclient.ini* datoteku, agent generira datoteku *nsclient.log*. U datoteci se detaljno upisuju greške rada agenta u jasno čitljivom tekstualnom formatu i ustanovio da se agent pokušava spojiti na nepostojeću *IP* adresu.

Sa strane Nagios Core poslužitelja koristili smo *widnows.cfg* konfiguracijsku datoteku koja služi kao template za povezivanje sa NSClient++ agentom koji se nalazi na našem računalu. Datoteka smo preimenovali u ime računala koje nadziremo i prilagodili za nadzor zauzeća diskovnog prostora, radne memorije i opterećenosti diska. Konfiguracija je prikazana u slici kod 4.1., a grafički prikaz na web sučelju izgleda ovako:

Slika 4.6 Grafički prikaz Windows uređaja

Kako se vidi iz slike 4.4 detaljno su objašnjene stavke koje nadziremo. Stupci prikazuju sljedeće, redom sa lijevo na desno:

- Ime računala
- Servis koji nadziremo

- Stanje servisa
- Vrijeme zadnje provjere
- Vrijeme neprekinutog rada
- Faze provjere
- Detaljan opis rezultata koje smo dobili

Za nadziranje mašine sa Linux operativnim sustavom koristimo *NRPE* agenta. Instalacija je jednostavna, ali ima nešto više konfiguriranja za razliku od NSClienta++:

- Instaliranje paketa koji su preduvjet za početak instalacije

```
yum install -y gcc glibc glibc-common openssl openssl-devel perl wget
```

- Skidanje paketa sa izvora

```
cd /tmp
wget --no-check-certificate -O nrpe.tar.gz https://github.com/NagiosEnterprises/nrpe/archive/nrpe-3.2.1.tar.gz
tar xzf nrpe.tar.gz
```

- Ako želimo prosljeđivati argumente preko NPRE –a moramo to specificirati u opciji konfiguracije

```
cd /tmp/nrpe-nrpe-3.2.1/
./configure --enable-command-args
make all
```

- Kreiranje nagios grupe i korisnika

```
make install-groups-users
```

- Instaliranje binarnih datoteka, *NRPE* pozadinskih procesa i *check_nrpe* dodatak

```
make install
```

- Instalacija konfiguracijskih datoteka

```
make install-config
```

- Nadogradanja servisnih datoteka */etc/services*

```
echo >> /etc/services
echo '# nagios services' >> /etc/services
echo 'nrpe 5666/tcp' >> /etc/services
```

- Instaliranje pozadinskih procesa

```
make install-init

systemctl enable nrpe.service
```

- Podešavanje vatrozida

```
firewall-cmd –zone=public –add-port=5666/tcp

firewall-cmd –zone=public –add-port=5666/tcp --permanent
```

- Pokrenemo servis i testiramo. Rezultat treba biti *NRPE v3.2.1*

```
systemctl start nrpe.service

/usr/local/nagios/libexec/check_nrpe –h 127.0.0.1
```

- Instalacija preduvjeta

```
yum install – y gcc blibc glibc-common make gettext automake autoconf wget openssl-devel net-snmp-utils epel-release

yum install –y perl net-snmp
```

- Instalacija dodataka sa izvora

```
cd /tmp

wget –no-check-certificate –o nagios-plugins.tar.gz https://github.com/nagios-plugins/nagios-plugins/archive/release-2.2.1.tar.gz

tar zxf nagios-plugins.tar.gz
```

- Instalacija

```
cd /tmp/nagios-plugins-release-2.2.1/

./tools/setup

./configure

Make

Make install
```

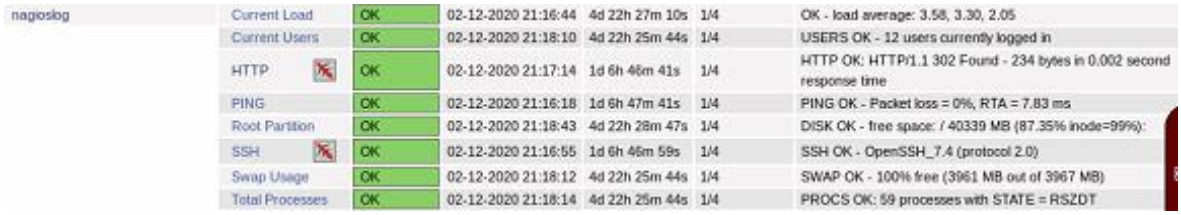
- Testiranje sa *check_load* naredbom

```
/usr/local/nagios/libexec/check_nrpe –H 127.0.0.1 –c check_load[8]
```

Za povezivanje *NRPE* agenta sa Nagios poslužiteljem koristimo konfiguracijsku datoteku localhost.cfg koja je svojevrsni template. Modificirali smo je za svoje potrebe, ali s obzirom

[8] https://support.nagios.com/kb/article.php?id=515

da je to jedina mašina, uz Nagios Core, ostavili smo sve opcije uključene. Konfiguracija je prikazana u slici kod 4.1., a grafički prikaz na web sučelju izgleda ovako:



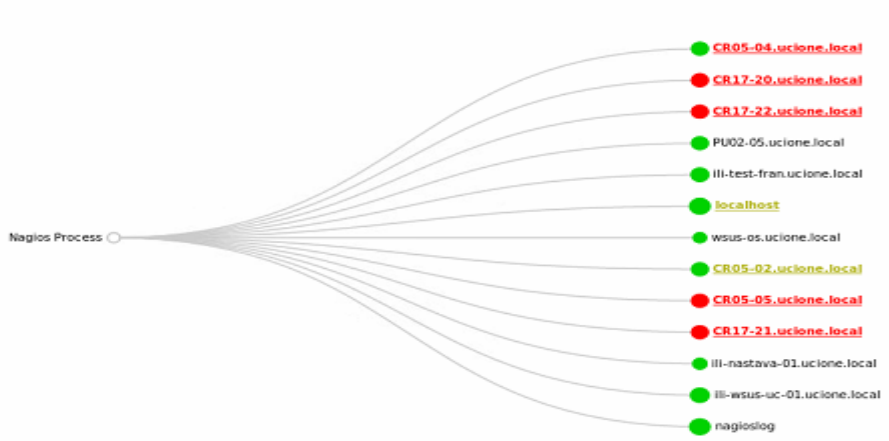
Slika 4.7 Grafički prikaz Linux uređaja

4.4. Generiranje izvještaja

Nagios Core posjeduje par odličnih načina generiranja izvještaja, ovisno o našim zahtjevima.

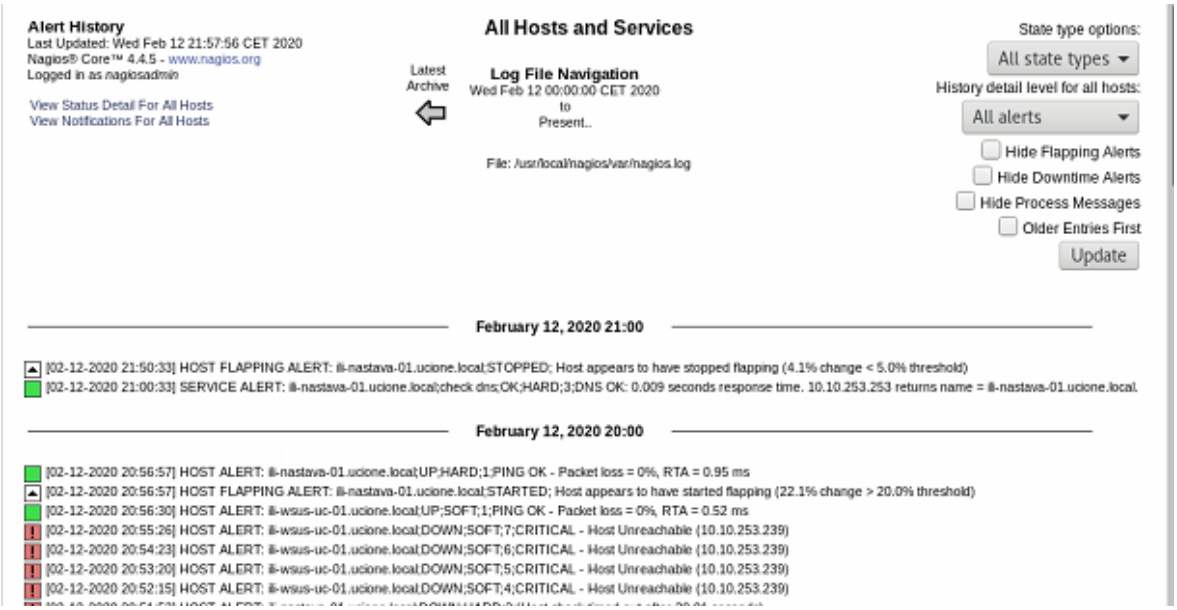
Slika 4.8 Strateško stanje

Ovaj prikaz nam pokazuje globalno zdravlje sustava. Generira se kroz web sučelje u stvarnom vremenu i ne možemo ga prebaciti u digitalni oblik osim kroz opciju ispisa cijelog ekrana.



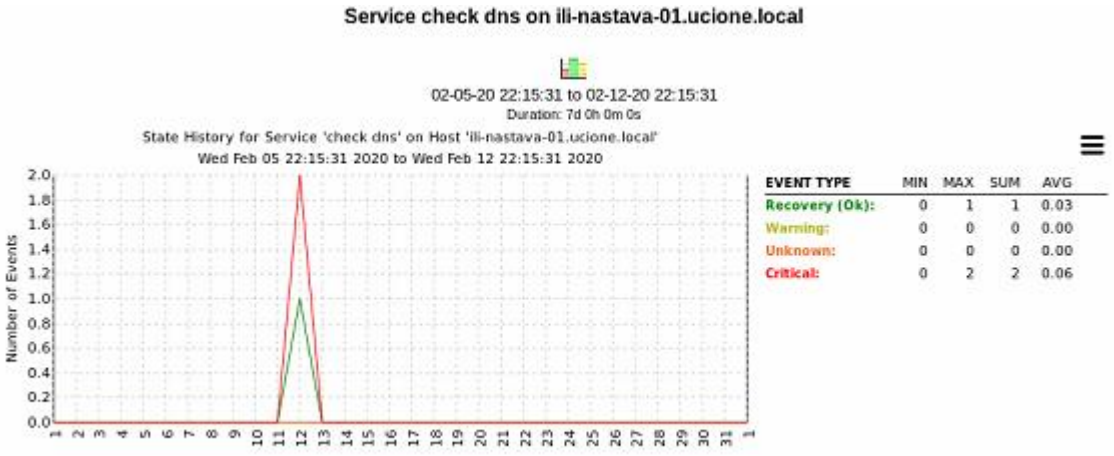
Slika 4.9 Mrežni prikaz uređaja

Na slici 4.7 vidimo interaktivnu kartu mrežne povezanosti Nagios Core-a i uređaja. Prelaskom miša preko bilo koje ikone ili naziva, možemo vidjeti status uređaja u stvarnom vremenu. Karta je prilagodljiva i pruža više načina pregleda.



Slika 4.10 Povijest upozorenja

Slika 4.8 prikazuje povijest upozorenja sa detaljnim opisom stanja. Iz ovoga možemo učiti da nam je jedan server bio u *flapping* stanju što upućuje na brze promjene kod servisa koji nadziremo. S obzirom da je pregledan i jasan na prvi pogled, ovaj izvještaj nam može koristi kad želimo na brzinu pogledati stanje upozorenja za taj dan.



Slika 4.11 Histogram

Slika 4.9 prikazuje jedini generator izvještaja u grafovima. Možemo specificirati za koji uređaj ili servis, koja stanja i u kojem vremenskom periodu. Odličan alat ako ste vizualni tip osobe i želite promatrati podatke kroz grafove.

Slika 4.12 Generiranje izvještaja dostupnosti servisa

Jedini generator izvještaja koji može izvoziti podatke u neku datoteku je izvještaj dostupnosti. Ima napredne mogućnosti generiranja izvještaja kroz stavke kao što su vremenski period, status servisa, početno stanje koje želimo mjeriti. Možemo ga izvesti u .csv datoteku, što nam daje mogućnost izvoza u program Excel i generiranje dodatnih grafova i prikaza.

Hostgroup 'linux-servers' Host State Breakdowns:

Host	% Time Up	% Time Down	% Time Unreachable	% Time Undetermined
localhost	100.000% (100.000%)	0.000% (0.000%)	0.000% (0.000%)	0.000%
nagioslog	78.624% (78.624%)	21.376% (21.376%)	0.000% (0.000%)	0.000%
Average	89.312% (89.312%)	10.688% (10.688%)	0.000% (0.000%)	0.000%

Hostgroup 'nastavaserveri' Host State Breakdowns:

Host	% Time Up	% Time Down	% Time Unreachable	% Time Undetermined
ili-nastava-01.ucione.local	99.926% (99.926%)	0.074% (0.074%)	0.000% (0.000%)	0.000%
Average	99.926% (99.926%)	0.074% (0.074%)	0.000% (0.000%)	0.000%

Hostgroup 'windows-servers' Host State Breakdowns:

Host	% Time Up	% Time Down	% Time Unreachable	% Time Undetermined
CR05-02.ucione.local	28.741% (39.545%)	43.938% (60.455%)	0.000% (0.000%)	27.321%
CR05-04.ucione.local	53.430% (72.819%)	19.944% (27.181%)	0.000% (0.000%)	26.626%
CR05-05.ucione.local	25.897% (35.724%)	46.594% (64.276%)	0.000% (0.000%)	27.510%
CR17-20.ucione.local	34.779% (49.104%)	36.049% (50.896%)	0.000% (0.000%)	29.172%
CR17-21.ucione.local	49.070% (57.653%)	36.043% (42.347%)	0.000% (0.000%)	14.887%
CR17-22.ucione.local	49.057% (57.638%)	36.056% (42.362%)	0.000% (0.000%)	14.887%
PU02-05.ucione.local	0.000% (0.000%)	0.000% (0.000%)	0.000% (0.000%)	100.000%
ili-test-fran.ucione.local	99.380% (99.981%)	0.019% (0.019%)	0.000% (0.000%)	0.601%
Average	42.544% (51.558%)	27.330% (35.942%)	0.000% (0.000%)	30.125%

Hostgroup 'wsusserveri' Host State Breakdowns:

Slika 4.13 Izvještaj dostupnosti

5. Testiranje

U ovom poglavlju opisane su metode kojima smo proveli testiranje, izvođenje testova u više vremenskih intervala i krajnje rezultate. Time smo pokazali razinu funkcionalnosti, isplativost sustava i njegovu svrhu.

5.1. Metode testiranja

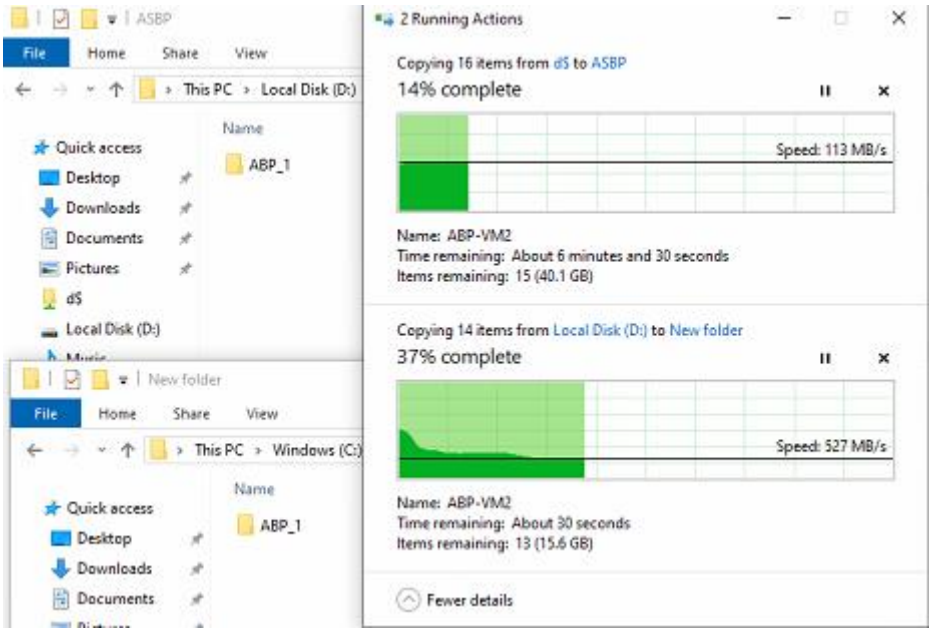
1. Za testiranje nadzora zapunjenosti diskovnog prostora kopirati ćemo i kasnije izbrisati veliku količinu podataka koja će izazvati *Warning, Critical* i *OK* stanje. Testirati ćemo na obje diskovne particije koje nadziremo. Naredba za *Warning* stanje će biti podešena na kapacitet slobodnog prostora.
2. Slanje obavijesti putem elektroničke pošte testirali smo dodavanjem novog računala.
3. Podešavanje konfiguracijske datoteke *contacts.cfg* koja nam diktira u kojem vremenskom periodu ćemo dobivati obavijesti servisa i po kojem kriteriju.

5.2. Provođenje testova

1. Na računalo CR05-02.ucione.local smo oslobodili dovoljno prostora da nam svi servisi budu u *OK* stanju što se vidi na slici 5.1.

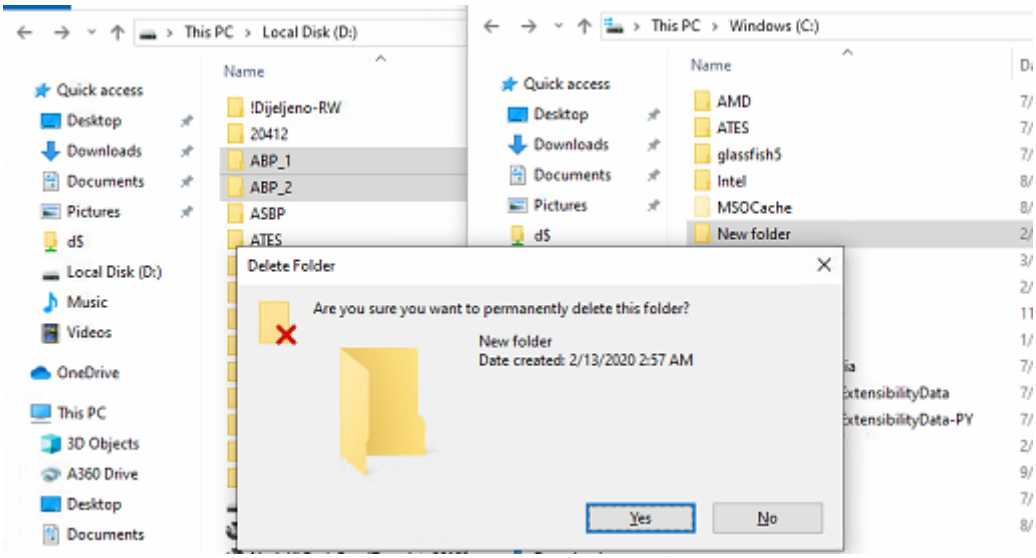
CR05-02.ucione.local	C:\ Drive Space	OK	02-13-2020 02:43:33	0d 0h 13m 21s	1/3	c: - total: 156.25 Gb - used: 126.37 Gb (81%) - free 29.88 Gb (19%)
	CPU Load	OK	02-13-2020 02:45:57	0d 13h 30m 57s	1/3	CPU Load 11% (5 min average)
	D:\ Drive Space	OK	02-13-2020 02:37:55	0d 13h 30m 49s	1/3	d: - total: 307.62 Gb - used: 226.65 Gb (74%) - free 80.97 Gb (26%)
	Memory Usage	OK	02-13-2020 02:46:10	0d 13h 30m 44s	1/3	Memory usage: total:18595.44 MB - used: 4486.46 MB (24%) - free: 14108.98 MB (76%)

Slika 5.1 CR05.02 Početno stanje računala CR05-02



Slika 5.2 Kopiranje datoteka

Slika 5.2 prikazuje kopiranje kopiranje virtualnih mašina koje koristimo za kolegije na visokoj školi.



Slika 5.3 Oslobađanje diskovnog prostora

Da bi oslobodili diskovni prostor morali smo obrisati mašine koje više nećemo koristiti.

2. Testiranje smo započeli tako što smo dodali novo računalo u Nagios, ali nismo izvršili resetiranje NSClient++ servisa na računalu.

PU02-05.ucione.local	C:\ Drive Space	CRITICAL	02-12-2020 09:05:35	0d 0h 26m 12s	3/3	CRITICAL - Socket timeout
	CPU Load	CRITICAL	02-12-2020 09:07:25	0d 0h 24m 22s	3/3	CRITICAL - Socket timeout
	D:\ Drive Space	CRITICAL	02-12-2020 09:09:15	0d 0h 22m 32s	3/3	CRITICAL - Socket timeout
	Memory Usage	CRITICAL	02-12-2020 09:11:06	0d 0h 20m 41s	3/3	CRITICAL - Socket timeout

Slika 5.4 Dodavanje računala PU02-05

Svi servisi su u stanju *Critical* jer Nagios ne može pristupiti podacima na računalu.

3. U konfiguracijskoj datoteci smo podesili da sva računala u *windows server host* grupi šalju obavijesti za vrijeme radnog vremena od 08:00 do 20:00 i da ne prijavljuju *Down* stanje. Ovom metodom trebali bi smanjiti količinu poruka elektroničke pošte, a da pritom ne ugrožavamo kvalitetu nadzora.

```
define timeperiod {

    name                    workhours
    timeperiod_name         workhours
    alias                   Normal Work Hours

    monday                  08:00-20:00
    tuesday                 08:00-20:00
    wednesday               08:00-20:00
    thursday                08:00-20:00
    friday                  08:00-20:00
    saturday                08:00-20:00
}
```

Slika 5.5 Radno vrijeme

```
define contact {

    name                            generic-contact         ; The name of this contact template
    service_notification_period     24x7                    ; service notifications can be sent anytime
    host_notification_period        24x7                    ; host notifications can be sent anytime
    service_notification_options    w,c,r,f,s               ; send notifications for all service states, flapping events, and sc
heduled downtime events
    host_notification_options       u,f,s                   ; send notifications for all host states, flapping events, and schedul
ed downtime events
    service_notification_commands   notify-service-by-email ; send service notifications via email
    host_notification_commands      notify-host-by-email    ; send host notifications via email
    register                        0                       ; DON'T REGISTER THIS DEFINITION - ITS NOT A REAL CONTACT, JUST A
TEMPLATE!
}
```

Slika 5.6 Micanje stanja *Down* i *Unknown*

5.3. Rezultati

1. Dobiveni rezultati upućuju na to da se stanje servis promijenilo, što dokazuje da web sučelje pokazuje točno stanje diskova.

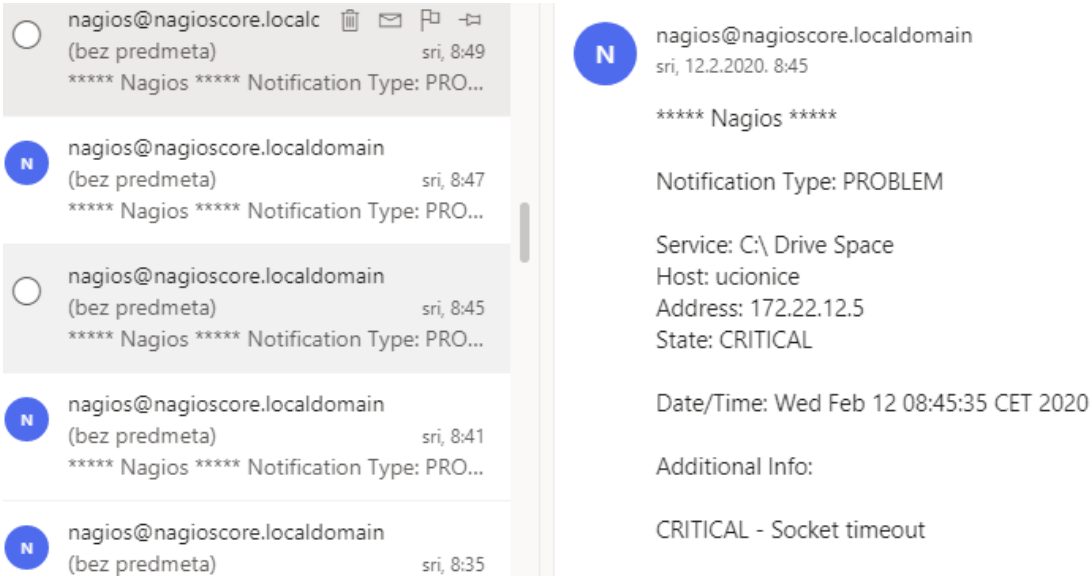
Host	Service	Status	Last Check	Duration	Attempt	Status Information
CR05-02.ucione.local	C:\ Drive Space	CRITICAL	02-13-2020 03:03:33	0d 0h 1m 39s	1/3	c: - total: 156.25 Gb - used: 151.49 Gb (97%) - free 4.76 Gb (3%)
	CPU Load	OK	02-13-2020 02:55:57	0d 13h 49m 15s	1/3	CPU Load 6% (5 min average)
	D:\ Drive Space	WARNING	02-13-2020 03:01:55	0d 0h 3m 17s	3/3	d: - total: 307.62 Gb - used: 273.39 Gb (89%) - free 34.22 Gb (11%)
	Memory Usage	OK	02-13-2020 02:56:10	0d 13h 49m 2s	1/3	Memory usage: total:18595.44 MB - used: 4670.25 MB (25%) - free: 13925.19 MB (75%)

Slika 5.7 Promjene na računalu CR05.02

CR05-02.ucione.local	C:\ Drive Space	OK	02-13-2020 03:17:33	0d 0h 4m 54s	1/3	c: - total: 156.25 Gb - used: 126.59 Gb (81%) - free 29.66 Gb (19%)
	CPU Load	OK	02-13-2020 03:15:57	0d 14h 6m 30s	1/3	CPU Load 2% (5 min average)
	D:\ Drive Space	OK	02-13-2020 03:21:55	0d 0h 0m 32s	1/3	d: - total: 307.62 Gb - used: 225.54 Gb (73%) - free 82.08 Gb (27%)
	Memory Usage	OK	02-13-2020 03:16:10	0d 14h 6m 17s	1/3	Memory usage: total:18595.44 MB - used: 4762.67 MB (26%) - free: 13832.77 MB (74%)

Slika 5.8 Računalo CR05-02 vraćeno u početno stanje nakon brisanja

2. Dobiveni rezultati upućuju da je sustav poslao upozorenje putem elektroničke pošte.



Slika 5.9 Obavijest elektroničke pošte

3. Rezultat testiranja su pokazali da se greške ne prijavljuju za vrijeme radnog vremena i u situacijama kada je uređaj ugašen. Poslije 20:00 nismo više primali obavijesti elektroničke pošte.

Zaključak

U okruženju u kojem je potrebno nadzirati veliku količinu mrežnih uređaja i različitih operacijskih sustava, Nagios Core se nametnu kao idealno rješenje. Instalacija nije komplicirana i popraćena je detaljnom dokumentacijom i velikim brojem entuzijasta koji aktivno sudjeluju u rješavanju problema na koja naiđete.

Isplativost se odmah na početku iskazuje u cijeni licenci i mogućnosti besplatne distribucije. Sama činjenica da je besplatan, dala nam je mogućnost da istražujemo i probamo mogućnosti koje nam se nude.

Alat se pokazao kao moćno sredstvo nadzora svojom agilnošću, mogućnostima nadzora i distribucijom na Windows operacijskim sustavu u našoj okolini koja je poprilično homogen. Možemo nadzirati veliku količinu mrežnih uređaja kroz instalaciju velikog broja agenta (ovisno o potrebama), kreiranjem predložaka i dobrom optimizacijom.

Web grafičko sučelje na kojem se prikazuju svi dodani uređaji, servisi i mogući izvještaji, pregledno je i ugodno korisniku. Potrebno je kratko vrijeme da se na njega naviknemo i počnemo ga aktivno koristiti.

Svaki sustav nadzora mora posjedovati precizan i dobar način slanja obavijesti administratorima sustava. Ovaj segment nam je bio najteže složiti, ali kada se osposobi jednostavno ga je koristiti i prilagoditi našim potrebama.

Nagios Core se pokazao kao kompletan sustav za nadzor, pokrio je sve bitne elemente koje smo namjeravali nadzirati, a u nekim dijelovima ih i nadmašio.

Popis kratica

.vhd	*virtual hard disk*	virtualni tvrdi disk
.vhdx	*virtual hard disk v2*	virtualni hard disk druge generacije
ISO	*ISO image*	datoteka koja sadrži cijelu sliku
IP	*Internet protocol*	protokol na internetu
HTML	*hypertext markup language*	standard za dizajn dokumenata
CFG	*configuration file*	konfiguracijska datoteka
FQDN	*fully qualified damain name*	ime računala na domeni
SMTP	*simple mail transfer protocol*	protokol za prijenos elektroničke pošte
NRPE	*nagios remote plugin executor*	Nagios agent za izvršavanje na daljinu
CR	*classroom*	učionica
LR	*lecture room*	dvorana
CSV	comma separated values	čisti tekst koji sadrži popis podataka

Popis slika

Slika 2.1.Prikaz TeamViewer obavijesti 9

Slika 3.1 Podešavanje Hyper-V okruženja 13

Slika 3.2 CentOS operacijski sustav 14

Slika 4.1 Nagios Core web sučelje 17

Slika 4.2 Direktor u kojem se nalaze konfiguracijske datoteke 19

Slika 4.3 Prikaz filtera događaja 22

Slika 4.4 Podešavanje naredbe za slanje elektroničke pošte u *commands.cfg* datoteci 24

Slika 4.5 Neuspješno spajanje Nagios-a i uređaja pomoću NSClinet++ agenta 25

Slika 4.6 Grafički prikaz Windows uređaja 26

Slika 4.7 Grafički prikaz Linux uređaja 29

Slika 4.8 Strateško stanje 29

Slika 4.9 Mrežni prikaz uređaja 30

Slika 4.10 Povijest upozorenja 30

Slika 4.11 Histogram 31

Slika 4.12 Generiranje izvještaja dostupnosti servisa 31

Slika 4.13 Izvještaj dostupnosti 32

Slika 5.1 CR05.02 Početno stanje računala CR05-02 33

Slika 5.2 Kopiranje datoteka 34

Slika 5.3 Oslobađanje diskovnog prostora 34

Slika 5.4 Dodavanje računala PU02-05 35

Slika 5.5 Radno vrijeme 35

Slika 5.6 Micanje stanja *Down* i *Unknown* 35

Slika 5.7 Promjene na računalu CR05.02 35

Slika 5.8 Računalo CR05-02 vraćeno u početno stanje nakon brisanja 36

Slika 5.9 Obavijest elektroničke pošte 36

Popis tablica

Tablica 4.1 Struktura direktorija i datoteka unutar Nagios Core-a..... 17

Tablica 4.2 Definicije za podešavanje Log datoteke..... 19

Popis kôdova

Kod 4.1 Konfiguracija host datoteke 20

Kod 4.2 *Nagios.cfg* datoteka 21

Literatura

[1] Tom Ryder. *Nagios Core Administration Cookbook second edition* : Packt Publishing Ltd, 2016.

[2] Wojciech Kocjan, Piotr Beltowski. *Learning Nagios – Third Edition*.: Packt Publishing Ltd, 2014.

[3] Wojciech Kocjan. *Learning Nagios 4*.: Packt Publishing Ltd, 2014.

[4] N. Guarracino*,†, V. Lavorini†, A. Tarasio†, ‡ And E. Tassi*,†; *Dipartimento Di Fisica, Università Della Calabria, Arcavacata Di Rende, Italy†Istituto Nazionale Di Fisica Nucleare, Gruppo Collegato Di Cosenza, Arcavacata Di Rende, Italy; High Performance Scientific Computing Using Distributed. *An Integrated Monitoring System,with Ganglia and Nagios,* Infrastructures Downloaded from www.worldscientific.com by UNIVERSITY OF MICHIGAN ANN ARBOR on 07/22/17. For personal use only

[5] J. Renita and N. Edna Elizabeth *Network's Server Monitoring and Analysis Using Nagios*; Electronics and communication Engineering, SSN College of Engineering, Kalavakkam, Chennai-603110

[6] V Fernandez, A Pazos, J Saborido and M Seco. *Arduino and Nagios integration for monitoring*. Instituto Galego de Fisica de Altas Enerxias (IGFAE), Universidad de Santiago de Compostela, Santiago de Compostela, Spain; OP Publishing Journal of Physics: Conference Series 513 (2014) 062015

[7] https://support.nagios.com/kb/article/nagios-core-installing-nagios-core-from-source-96.html#CentOS, 01.12.2019

[8] https://support.nagios.com/kb/article/nrpe-how-to-install-nrpe-8.html, 05.12.2019

[9] https://support.nagios.com/kb/article.php?id=515, 01.12.2019

[10] https://assets.nagios.com/downloads/nagios-log-server/docs/Sending-Nagios-Core-Logs-To-Nagios-Log-Server.pdf, 01.01.2020

[11] https://www.youtube.com/watch?v=zMBNdPBAMyA, 27.12.3019

[12] https://www.youtube.com/watch?v=ksCfyyHj3iA, 10.01.2020

[13] https://support.nagios.com/kb/category.php, 01.12.2019

ALGEBRA
VISOKO
UČILIŠTE

Nagios Core server za nadzor IT sustava

Pristupnik: Mladen Plećaš, 1192010530

Mentor: predavač Zlatan Morić