AMPEC

Академска мрежа Србије

Препоруке безбедности мрежних сервиса

Историја верзија документа

Верзија	Датум	Иницијали аутора	Опис промене
1.0	2009. год	Душан Пајин (РЦУБ), Ратко Бучић (ЈУНИС), Владимир Илић (АРМУНС)	Прва верзија документа у оквиру AMPEC пројекта
2.0	05.08.2015.	Милош Куколеча (AMPEC)	Друга верзија документа

Садржај

1 Увод

Овај документ треба да дефинише препоруке заштите и безбедности мрежних инфраструктурних сервиса. У ову групу сервиса спадају сви сервиси који треба да обезбеде непрекидно и сигурно функционисање рачунарске мреже.

У наставку документа обрађени су следећи сервиси:

- Сервис за логовање системских порука (*Syslog*)
- Сервис за синхронизацију времена (NTP)
- Сервис за надгледање и управљање уређајима (*SNMP*)

2 Сервис за логовање системских порука (*Syslog*)

Сви рачунарски системи користе принц*IP* чувања системских порука о одређеним догађајима и њихов преглед. На неким системима се овакве апликације зову *Event Viewer, Event Log* или *Syslog*. Мрежни уређаји и сервери ове поруке могу чувати у радној меморији уређаја или неком фајлу, а чест случај је и њихово слање преко мреже.

Syslog је стандардизовани протокол за слање лог информација преко мреже. Потиче из доба *Unix* система и омогућава да се информације о одређеним догађајима на систему пренесу преко мреже до централизованог сервера за прикупљање оваквих информација. Централизовани систем омогућава лаку претрагу и преглед информација о дешавањима на више различитих система на мрежи.

Овакав начин централизованог чувања информација о догађајима на рачунарским системима и мрежним уређајима је од посебног значаја за сигурност мреже. Сигурност мреже може бити повећана анализом информација о догађајима у мрежи након неког безбедносног проблема, његовом детекцијом и анализом, што ће допринети избегавању сличног проблема у будућности. Са друге стране, мрежни уређаји могу послати *Syslog* поруку у случају детекције неког напада или кршења одређених безбедносних правила. *Syslog* је једини начин на који мрежни уређај може брзо да обавести администраторе у мрежи да постоји одређени проблем. Ово је изузетно битно у случају када постоји велики број уређаја у мрежи.

Syslog је протокол апликативног слоја који служи за слање системских порука и ради на клијент-сервер принц*IP*у. *Syslog* сервер је апликација која прима *Syslog* поруке и чува их на систему, на пример, у фајлу или у бази података. *Syslog* клијент је уређај који шаље своје системске поруке ка *Syslog* серверу. *Syslog* поруке се преносе се преко транспортног протокола *UDP* на порту 514.

Syslog поруком се добија пет информација од значаја:

- Уређај који је генерисао поруку
- Време када је порука генерисана (*Timestamp*)
- Текст поруке
- Објекат на који се шаље порука (*Facility*)
- Ниво критичности поруке (*Severity*)

Информација о уређају који је генерисао поруку добија се из изворишне адресе *IP* пакета који носи *Syslog* поруку и не налази се у склопу *Syslog* поруке. Време када је порука генерисана ја изузетно значајна информација, како би се одредио редослед одређених догађаја у мрежи. Обзиром да један догађај може утицати на више уређаја у мрежи и сваки од њих може генерисати *Syslog* поруку на основу тог догађаја, изузетно је значајно да уређаји имају усклађено време како би поруке на централизованом уређају могле да буду сврстане у правилном редоследу. За синхронизацију времена на уређајима користити *NTP* протокол. Текст поруке је дефинисан конкретним догађајем и представља кратак опис. Објекат на који се шаље порука може да дефинише различите локације на које ће поруке бити снимљене (на пример, различите фајлове), а за мрежне уређаје се најчешће користи *facility* од *local0* до *local7*. Ниво критичности дефинисан је у распону бројева од 0 до 7, према принципу да је мањи број критичнији. Сваком нивоу критичности одговара одређени назив који је приказан у табели.

Табела 1 – Нивои критичности Syslog порука

Број	Назив
0	*Emergency*
1	*Alert*
2	*Critical*
3	*Error*
4	*Warning*

5	*Notice*
6	*Informational*
7	*Debug*

2.1 Имплементација логовања системских порука са мрежних уређаја

За логовање системских порука преко *Syslog* протокола потребан је сервер са апликацијом која ће примати поруке.

На тржишту постоје различите апликације за скупљање *syslog* порука, од бесплатних до комерцијалних које раде и на *Windows* и на *Linux* оперативним системима. У оквиру сваке *Linux* дистрибуције постоји апликација која може да има улогу *syslog* сервера и потпуно је бесплатна. Мана ове апликације јесте што се поруке снимају у текстуални фајл и не постоји напредни систем за претраживање порука, па се њихово претраживање своди на претраживање текстуалног фајла што није скалабилно решење.

Апликација за чување *syslog* порука треба да омогући брзо и ефикасно претраживање *syslog* порука обзиром да у већим мрежама број порука може бити изузетно велики. Претраживање мора бити могуће према временском интервалу у коме је порука генерисана, нивоу критичности, уређају који је генерисао поруку и према произвољном стрингу унутар текста поруке.

AMPEC *CSIRT* и хелпдеск тимови користи *NetIIS* систем за примање, чување и претраживање системских порука.

У наставку је наведена конфигурација мрежних уређаја за коришћење и слање *syslog* порука. *Syslog* поруке се шаљу ка *NetIIS* серверу који је на адреси 147.91.3.12. Ка серверу се шаљу поруке нивоа критичности од 0 до 7. Препоручује се да изворишна адреса порука буде адреса *Loopback0* интерфејса који је увек активан и са којом ће мрежни уређај бити идентификован на серверу. Такође, препорука је да постоји инверзни *DNS* унос за *IP* адресу *Loopback0* интерфејса како би идентификација уређаја могла да се уради и према имену.

Препоручује се да се системске поруке се чувају и у меморији мрежног уређаја која функционише по принципу кружног бафера и користи 100 килобајта системске меморије. Овакав принцип се користи у случају да, на пример, *syslog* сервер није доступан или је потребно видети задње поруке директно на уређају што је корисно у свакодневном раду и при промени конфигурације уређаја. У меморију се логују сви нивои критичности, па и излаз из *debug* команди. Додатно, системске поруке се шаљу и на терминалске линије, али се не шаљу на конзолну линију.

```
ImeRutera(config)#logging on
ImeRutera(config)#logging host 147..91.3.12
ImeRutera(config)#logging trap informational
ImeRutera(config)#logging source-interface Loopback0
ImeRutera(config)#logging buffered 100000
ImeRutera(config)#logging buffered debug
ImeRutera(config)#logging monitor informational
ImeRutera(config)#no logging console
```

3 Сервис за синхронизацију времена (*NTP*)

Одређени безбедносни инцидент може утицати на више уређаја у мрежи у зависности од типа инцидента и величине мреже. Сваки уређај може генерисати *syslog* поруку на основу наведеног инцидента, па је и изузетно значајно да уређаји имају усклађено време како би поруке на централизованом уре ђају могле да буду сврстане у правилном редоследу. У случају да време на уређајима није синхронизовано, не можемо утврдити тачан редослед догађаја као ни апсолутно време у којем се одиграо наведени инцидент.

NTP (Network Time Protocol) се користи као сервис за синхронизацију часовника на компјутерским системима преко пакетских мрежа са променљивим кашњењем. Последња дефинисана верзија кроз *IETF* стандард је верзија 3. *NTP* користи *UDP* као транспортни протокол на предефинисаном порту 123.

У погледу прецизности времена користи се хијерархијска структура у виду слојева (Стратум), где слојеви дефинишу удаљеност од референтног извора тачног времена:

- Слој 0 - представљају уређаји који одређују "тачно време", као што су атомски часовници, *GPS* и радио часовници. Ови уређаји су преко одређеног интерфејса (најчешће *RS-232*) везани на рачунар.
- Слој 1 - представљају уређаји који су директно везани на уређаје из слоја 0. Ови рачунари играју улогу *NTP* сервера и везани су на рачунарску мрежу.
- Слој 2 - представљају уређаји који се синхронизују на *NTP* серверима слоја 1. Уређаји слоја 2 уједно представљају *NTP* сервере за уређаје који се са њима синхронизују и припадају слоју 3.

На овај начин постигнута је хијерархијска структура *NTP* сервера и њихове релативне тачности. Синхронизација уређаја у мрежи је значајна и одређени сервиси као што су *syslog* поруке, дигитални сертификати или листе приступа базиране на времену директно зависе од међусобне синхронизације уређаја у мрежи. Напад на *NTP* сервере или евентуална несинхронизованост уређаја може довести до непрецизних логова на системима или престанка рада одређених функција. Из наведених разлога треба обратити посебну пажњу на заштиту *NTP* протокола на мрежних уређајима.

3.1 Имплементација *NTP* протокола

У наставку је приказана конфигурација *NTP* синхронизације коју треба применити на мрежним уређајима AMPEC корисника. Добра пракса је синхронизација централних мрежних уређаја у мрежи AMPEC корисника са екстерним *NTP* серверима којима се може веровати. Пожељна је употреба кључа за синхронизацију уколико је то могуће. Остали мрежни уређаји у мрежи AMPEC корисника се потом могу синхронизовати са централним уређајима.

Први пример је конфигурација *NTP* централних уређаја у мрежи AMPEC корисника. Листе приступа 11 дефинишу са којим *IP* адресама је дозвољена синхронизација, док листа 12 дефинише адресе којима се допушта да се синхронизују.

```
! Centralni uredjaji u mrezi
!
clock timezone CET 1
clock summer-time CET recurring last Sun Mar 2:00 last Sun Oct 2:00
!
ntp authenticate
ntp authentication-key 1 md5 <kljuc1> ntp trusted-key 1
ntp authentication-key 2 md5 <kljuc2>
```

```
!
ntp source loopback 0
ntp server ntp.rcub.bg.ac.rs key 1
prefer ntp server ntp.phy.bg.ac.rs
ntp access-group peer 11
access-list 11 permit host 147.91.1.x
access-list 11 permit host 147.91.87.182
!
ntp access-group serve-only 12
!
access-list 12 permit 147.91.0.0   0.0.7.255
```

Други пример представља конфигурацију мрежног уређаја који није централни уређај у мрежи AMPEC корисника. За аутентификацију се користи кључ бр. 2.

```
!
! Ostali mrezni uredjaji
!
clock timezone CET 1
clock summer-time CET recurring last Sun Mar 2:00 last Sun Oct 2:00
!
ntp authenticate
ntp authentication-key 2 md5 <kljuc2>
ntp trusted-key 2
!
ntp source loopback 0
ntp server 147.91.0.112 key 1
ntp server 147.91.0.124 key 1
ntp access-group peer 11
!
access-list 11 permit host 147.91.0.112
access-list 11 permit host 147.91.0.124
```

4 Сервис за надгледање и управљање уређајима (*SNMP*)

Сервис надгледања уређаја је користан не само у делу који се тиче функционисања рачунарских мрежа и детектовања прекида, већ може указати на безбедносне проблеме и детектовати инциденте. Основни протокол апликативног слоја који се користи за надгледање и управљање мрежним уређајима је *SNMP* (*Simple Network Management Protocol*). *SNMP* користи *UDP* као транспортни протокол на порту 161, док се посебне *SNMP* трап поруке преносе преко *UDP* порта 162.

Највећи део функционалности система за надгледање мреже ослања се на *SNMP* протокол. У најчешћем сценарију коришћења *SNMP* протокола, у мрежи постоји један сервер за надгледање и управљање (*NMS – Network Management Station*) и већи број уређаја који се надгледају и којима се управља. Сваки од уређаја који се надгледа или управља поседује софтверску компоненту која се назива *SNMP* агент и која преко *SNMP*-а комуницира са сервером. *SNMP* агент има за циљ да учини доступним различите корисне податке који могу да се надгледају, на пример заузеће процесора и меморије, број процеса или статус мрежних интерфејса. Ове податке у регуларним интервалима времена прикупља и анализира сервер за надгледање мреже преко *SNMP* променљивих које су означене идентификаторима (*OID – Object Identifier*). Скуп променљивих које могу да се прате на уређајима дефинисан је кроз базу *SNMP* променљивих (*MIB – Management Information Base*).

Са друге стране, недовољно заштићен *SNMP* приступ одређеном уређају представља велику сигурносну претњу. Као што *NMS* може да прикупи велики број корисних информација са уређаја, тако и потенцијални нападач у случају да има приступ уређајима посредством *SNMP* протокола може прикупити корисне информације које ће искористити за касније нападе у мрежи.

Уобичајена имплементација *SNMP* агената на мрежним уређајима дозвољава два начина приступа од којих је једним дозвољено само читање *SNMP* променљивих (*read-only*), док је другим дозвољено читање и промена вредности *SNMP* променљивих (*read-write*).

SNMP протокол је дефинисан у три верзије које се донекле разликују по одређеним типовима команди, али најзначајнија разлика се огледа у безбедносним карактеристикама протокола.

SNMPv1 и *SNMPv2c* користе једноставан принцип аутентификације заснован на некој врсти лозинке, која се зове "*Community String*". То конкретно значи да је за читање или постављање *SNMP* променљивих на неком уређају потребно имати исти *Community* параметар који је дефинисан на *SNMP* агенту уређаја. Додатни проблем представља чињеница да се *SNMP Community* параметар кроз мрежу преноси у незаштићеном текстуалном облику, па је прислушкивање пакета посебан безбедносни проблем *SNMP* протокола. Такође, *SNMP Community* параметар је као и свака врста лозинке подложан *brute-force* и *dictionary* нападима. Ове две верзије не имплементирају било какав додатни систем аутентификације или енкрипције пакета, па су безбедносни проблеми главни разлог дефинисања новије верзије протокола, означене као *SNMPv3*.

SNMPv3 дефинише додатну безбедност *SNMP* протокола кроз опциону аутентификацију *SNMP* пакета коришћењем *MD5* или *SHA1 hash* функција и опциону енкрипцију на бази *DES* алгоритма. Иако је очигледно да *SNMPv3* решава безбедносне проблеме претходних верзија *SNMP* протокола, *SNMPv3* је ретко имплементиран на системима за надгледање и управљање мрежом, али и на крајњим мрежним уређајима.

4.1 Имплементација *SNMP* протокола у AMPEC мрежи

Имајући у виду претходно наведене опште и безбедносне карактеристике *SNMP* протокола, за коришћење се генерално препоручује *SNMPv3* са укљученом аутентификацијом и енкрипцијом *SNMP* пакета. У случају да систем за надгледање и управљање мрежом који се користи унутар мреже не подржава *SNMPv3*, треба користити *SNMPv2c* са посебним мерама заштите.

Конкретно у Академској мрежи се као систем за надгледање мреже користи *NetIIS*, који тренутно подржава само *SNMPv1* и *SNMPv2c*. Са друге стране, већина мрежне опреме

подржава *SNMPv3*. AMPEC *CSIRT* тим препоручује имплементацију подршке за *SNMPv3* где год је то могуће, како би се повећала безбедност *SNMP* протокола на мрежним.

За тренутно коришћење *SNMP*, за потребе надгледања мреже и сервера, ЦСИРТ тим наглашава да је потребно конфигурацију *SNMPv2c* посебно заштитити. Дефинисати *SNMP Community* параметар са најмање 12 карактера коришћењем препорука за креирање лозинки. Ограничити *SNMP* приступ са одређених *IP* адреса, или само са *IP* адресе система за управљање и надгледање мреже. Користити приступ који омогућава само читање података (*read-only*).

У наставку је наведена конфигурација мрежних уређаја за *SNMPv2c*. Конфигурисан је *Community* параметар са искључиво приступом за читање *SNMP* променљивих, а приступ је ограничен само са адреса сервера за надгледање (*IP* адреса 147.91.3.12 у примеру). Укључено је слање *SNMP trap* порука ка серверу за надлгедање.

```
snmp-server community <Community> RO snmp-in
snmp-server contact AMRES Helpdesk helpdesk@amres.ac.rs
snmp-server location RCUB
!
snmp-server enable traps snmp
snmp-server trap link ietf
snmp-server trap-source Loopback 0
!
snmp-server host 147.91.3.12 traps version 2c <Community>
!
ip access-list standard snmp-in
 permit 147.91.3.8 0.0.0.7
 permit 147.91.4.8 0.0.0.7
```

У случају коришћења протокола *SNMPv3*, треба имплементирати конфигурацију приказану у наставку. За коришћење *SNMPv3* потребно је конфигурисати корисника који ће користити кључеве „лозинка1" за аутентификацију и „лозинка2" за енкрипцију. Приступ *SNMP* протоколом је према дефинисаној листи приступа дозвољен само са адреса сервера за надгледање (*IP* адреса 147.91.3.12 у примеру). Укључено је слање *SNMP trap* порука према серверу са *IP* адресе *Loopback0* интерфејса рутера.

```
snmp-server group snmp-grupa v3 auth priv
!
snmp-server user <korisnik> snmp-grupa v3 auth md5 <lozinka1> priv des56
<lozinka2> access snmp-in
!
snmp-server enable traps snmp
!
snmp-server source-interface traps Loopback 0
snmp-server host 147.91.3.12 traps version 3 priv <community>
```

```
ip access-list standard snmp-in
 permit 147.91.3.8 0.0.0.7
 permit 147.91.4.8 0.0.0.7
```