Pécsi Tudományegyetem
Pollack Mihály Műszaki Kar
Mérnök Informatikus Szak

SZAKDOLGOZAT

Naplózó rendszerek összehasonlítása

Készítette: Jákli Bence Márk
Témavezető: Iványi Péter
Pécs

2011

PÉCSI TUDOMÁNYEGYETEM
POLLACK MIHÁLY MŰSZAKI KAR

Szakdolgozat száma:

Mérnök Informatikus Szak

MIB II.-83/2007/2011.

SZAKDOLGOZAT FELADAT

Jákli Bence Márk

.....

hallgató részére

A záróvizsgát megelőzően szakdolgozatot kell benyújtania, amelynek témáját és feladatait az alábbiak szerint határozom meg:

Téma:

Naplózó rendszerek összehasonlítása

Feladat:

- A naplózó rendszerrrel szemben támasztott követelmények ismertetése
- Novell Sentinel Log Manager szoftver telepítése és tesztelése
- Syslog-ng szoftver telepítése és tesztelése
- A két rendszer összehasonlítása

A szakdolgozat készítéséért felelős tanszék: Rendszer és Szoftvertechnológia Tanszék

Külső konzulens: ……
munkahelye:…….

Témavezető: Iványi Péter
munkahelye: PTE-PMMK Rendszer és Szoftvertechnológia Tanszék

Pécs, 2011. február 10.

Dr.Szakonyi Lajos
op. szakvezető

HALLGATÓI NYILATKOZAT

Alulírott szigorló hallgató kijelentem, hogy a szakdolgozat saját munkám eredménye. A felhasznált szakirodalmat és eszközöket azonosíthatóan közöltem. Egyéb jelentősebb segítséget nem vettem igénybe.
Az elkészült szakdolgozatban talált eredményeket a főiskola, a feladatot kiíró intézmény saját céljaira térítés nélkül felhasználhatja.

Pécs, 2011. június 10.

.....
hallgató aláírása

Tartalomjegyzék

1. Bevezetés

1.1. Célok

Az utóbbi időben egyre nagyobb hangsúlyt kapott az informatika területén a biztonság. Ez köszönhető többek között az informatikai eszközök és az internet széles körű elterjedésének. Ma már gyakorlatilag nélkülözhetetlenné vált az informatikai háttér a vállalatok számára, melynek számos előnye van. Az informatikai rendszerek tervezésénél nagy hangsúlyt kell fektetni a biztonság kialakítására, mivel az informatikai bűnözés komoly károkat tud okozni, és veszélyt jelent az üzleti folyamatokra. Az informatikai biztonság egyik alappillére a megfelelő naplózási rendszer.

A naplózás segítségével azonban nem csak a biztonság növelhető. Jól kialakított naplózó rendszer segítségével átfogó képet kapunk a rendszerről, így megfigyelhető, elemezhető a rendszer működése, könnyen felismerhetők a rendszer esetleges hibái, gyengeségei.

A szakdolgozatom célja, hogy bemutassam a jelenleg rendelkezésre álló, leggyakrabban használt naplózó rendszereket, melyek alkalmasak nagyobb rendszerek naplózására. A dolgozatban nem fogok kitérni a naplózó rendszer kiépítésének és telepítésének minden egyes részletére, mivel a dolgozat fő célja az összehasonlítás, és a telepítés részletes leírása egy rendszer esetén is átlépi egy szakdolgozat terjedelmét.

1.2. Mit nevezünk naplózásnak, és miért van rá szükség

Naplózásnak nevezzük, amikor a rendszer bizonyos eseményeit rögzítjük. Ilyen esemény lehet például, ha egy felhasználó bejelentkezik, vagy egy folyamat futni kezd a rendszeren, de gyakorlatilag bármilyen esemény rögzíthető, ami a rendszeren történik. Különböző rendszerek különböző szintű eseményeket rögzítenek a beállításaiktól függően. Egy naplófájl bejegyzésekből áll, melyeket a különböző programok, és az operációs rendszer folyamatai készítenek. A bejegyzések formátuma eltérő lehet, UNIX alapú rendszereken általában a következő adatokat tartalmazza:

- Időbélyeg (Az üzenet létrehozásának ideje. Sokféle időbélyeg formátum létezik.)
- Hostname (A rendszer neve, amelyről az üzenet származik. Központosított naplózás esetén van szerepe.)
- Folyamatnév (A folyamat neve, amely a bejegyzést létrehozta.)
- PID (Process ID - folyamat azonosító. Mivel azonos névvel több folyamat is futhat, mindegyik rendelkezik egy egyedi azonosítóval.)
- Üzenet (az üzenet szövege)

A naplózás célja lehet például a rendszer optimalizálása, hibajavítás, statisztika készítés, vagy akár a felhasználók aktivitásának figyelése. Amennyiben egy rendszer eseményeit megfelelően naplózzuk, egy esetleges hiba esetén sokat segíthet a javításban, mivel vannak információink a rendszer állapotáról a hiba bekövetkezésének idejéről.
A naplózás feladatai közül egyre inkább előtérbe kerül a biztonság növelése. Amint egy számítógépet hálózatra kötünk, kitesszük a hálózat felől érkező támadásoknak. A naplózás segít abban, hogy ezeket a támadásokat időben észrevegyük, vagy egy esetleges betörés esetén pontosan tudjuk, hogy mely fájlokhoz férhetett hozzá a betörő, illetve a rendszer elemzésével megszüntethetjük a biztonsági rést, így később nem fordulhat elő hasonló behatolás.
Mivel gyakorlatilag bármilyen esemény naplózható, a naplózás számos más lehetőséget rejt magában.

1.3. Napló kezelési modellek

Többféle napló menedzsment modell létezik. A legegyszerűbb, amikor a naplófájlokat azon a gépen tároljuk és dolgozzuk fel, amelyiken készülnek.
Amikor egynél több eszköz naplózására van szükségünk, mindenképpen egy központosított megoldást célszerű választani. Egyszerű megoldás, ha bizonyos időközönként egy hálózati tárolóra vagy egy távoli számítógépre másoljuk az adatokat, azonban ezzel a módszerrel nem kapunk képet a rendszer aktuális állapotáról, és a naplófájlokat sok gép esetén nehéz rendszerezni.

A legjobb megoldás, ha egy központi szervert használunk, amely egy előre megadott porton hallgatózva fogadja a naplóüzeneteket a kliensektől, és egy központi adatbázisban tárolja a kapott adatokat, ahogy az az 1.1-es ábrán is látható. Ilyen központosított naplózó rendszer a Syslog-ng és a Novell Sentinel Log Manager.

1.1 ábra Központi naplókezelés

1.4. Naplófájlok biztonsága

A naplófájlok biztonsága nagyon fontos tényező egy rendszer tervezésénél. A naplófájloknak a következő feltételeknek mindenképpen meg kell felelniük:

1.4.1. Hozzáférés korlátozás

A naplófájl a legtöbb esetben értékét veszti, ha minden felhasználó jogosult rá, hogy hozzáférjen. Ha valaki be tud törni a rendszerünkbe, majd törli a naplófájlokból a betörésre utaló bejegyzéseket, vagy egyszerűen az egész naplófájlt, nem fogjuk megtudni a betörés részleteit. A naplófájlokban gyakran olyan információk is szerepelhetnek, amely nem tartozik minden felhasználóra, ezért célszerű az olvasási jogot is minimálisra korlátozni.

1.4.2. Átvitel megbízhatósága

Az adatokat el kell juttatnuk a központi szerverre a hálózaton keresztül. A első központi naplózó rendszerek az UDP protokollt használták az adatok szállítására. Az UDP azért nem jó választás, mert nem megbízható protokoll. Az adatok elveszhetnek a küldés során, és erről egyik fél sem értesül. Jobb döntés, ha a naplóbejegyzéseket TCP protokollon keresztül továbbítjuk. A TCP protokoll tartalmaz ellenőrzést, így biztosak lehetünk benne, hogy minden adat megérkezik. Ennek a protokollnak a legfőbb hátránya, hogy titkosítatlan adatokat szállít, így az adatok lehallgathatók, vagy akár módosíthatók is az átvitel során.

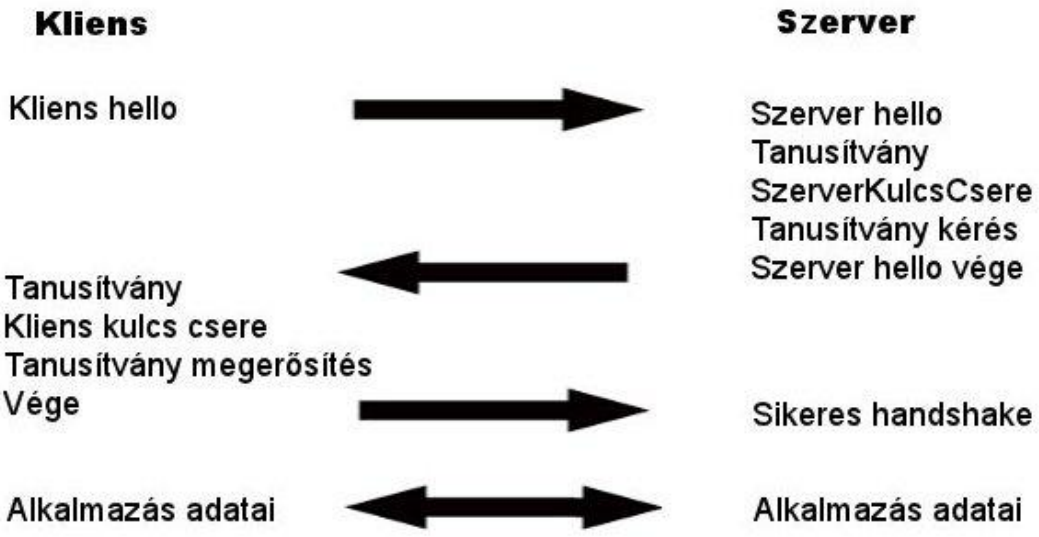
1.4.3. Az adatok biztonsága(SSL/TLS)

Ha TCP protokollt használunk az átvitelhez, akkor biztosított az adatok megérkezése. Azonban nem lehetünk biztosak benne, hogy kitől jött az adat, nem módosították-e, vagy esetleg nem hallgatták-e le az átvitel során. Erre kínál jó megoldást az SSL használata. Az SSL a Secure Sockets Layer-biztonságos csatlakozó réteg rövidítése. Ezt a protokollt a Netscape Communications Corp. Fejlesztette, majd szabványosítás céljából átadta az IETF-nek, ettől kezdve TLS (Transport Layer Security) lett a neve [2]. Az SSL/TLS a szállítási és az alkalmazási réteg között helyezkedik el, és biztonságos kommunikációt biztosít egyetlen csatornán. A kommunikációhoz természetesen mindkét oldalon szükség van a támogatására, azonban a felette lévő alkalmazások számára transzparens. Tartalmaz hitelesítést, így biztosak lehetünk benne, hogy az adat a megfelelő helyről érkezik. Az adatokat titkosított csatornán szállítja, így megőrizhető az adatintegritás, és nem kell tartanunk attól, hogy valaki lehallgatja a csatornát. A TLS a hitelesítéshez X.509 tanusítványokat használ. Az X.509-es tanusítványok formátuma az ITU-T nemzetközi szervezet által kiadott szabványon alapul, és a következő elemeket tartalmazza[3]:

- Tanusítvány
  - Verzió
  - Sorozatszám
  - Algoritmus azonosító
  - Kiállító
  - Érvényesség

        - Dátum amely előtt nem érvényes
        - Dátum amely után nem érvényes
    - Tárgy
    - Nyilvános kulcs információk
        - Algoritmus
        - RSA nyilvános kulcs
    - Kiegészítés
- Tanusítvány aláíró algoritmus
- Tanusítvány aláírás

A tanusítványok létrehozása különböző operációs rendszerek esetén eltérő lehet, azonban nem függ a használt naplózó rendszertől. A teljes szabványleírás elérhető az ITU-T oldalán, a http://www.itu.int/rec/T-REC-X.509/en címen. A tanusítványok használatakor fokozottan ügyeljünk rá, hogy a lejárat előtt megújítsuk őket.



1.2 ábra TLS kapcsolat felépítése (RFC 5246 alapján)

A biztonságos kommunikációs csatornát felépítő folyamat lépései az 1.2-es ábrán láthatók. A hello üzenetekkel a szerver és a kliens először egyeztetik a

kommunikációhoz szükséges alapvető változókat, mint például a verziószám, és a használt tikosítási és tömörítési eljárás. Ezután megtörténik a tanusítványok átvitele, illetve az authentikáció. Miután a kapcsolat felépült, a szerver biztos lehet benne, hogy az üzenetek a klienstől érkeznek, a kliens pedig biztos lehet benne, hogy az üzeneteit csak a szerver olvashatja el. A TLS 1.2-es verziójának leírását az RFC 5246-os dokumentum tartalmazza, amely a http://tools.ietf.org/html/rfc5246 címen érhető el. A syslog és a TLS összekapcsolását az RFC 5425-ös dokumentum tartalmazza, amely a http://tools.ietf.org/html/rfc5246 címen érhető el.

1.4.4. Tűzfal beállítása

Bármilyen rendszert építünk ki, mielőtt a hálózatra kötjük, a legfontosabb lépés, hogy állítsuk be a megfelelő tűzfal szabályokat. A tűzfal beállításait linux alapú rendszereken általában az iptables parancs segítségével kezelhetjük. A tűzfal beállításánál azt az alapelvet célszerű követni, hogy mindent tiltunk, kivéve, ami feltétlenül szükséges a rendszer működéséhez. A következő példakonfigurációban minden forgalmat letiltok, kivéve az ssh és a syslog kommunikációt.

```
#minden szabály törlése
iptables -F
iptables -X
#3 alapvető lánc létrehozás
iptables -P INPUT DROP
iptables -P OUTPUT DROP
iptables -P FORWARD DROP
#loopback kommunikáció engedélyezése
iptables -A INPUT -i lo -j ACCEPT
iptables -A OUTPUT -o lo -j ACCEPT
```

```
#syslog bejövő kapcsolatok engedélyezése, az általunk használt IP tartományból

iptables -A INPUT -p tcp -s 91.83.45.0/24 -d 91.83.45.77 --sport 513:65535 --dport
1514 -j ACCEPT


#ssh engedélyezése
 az általunk használt IP tartományból
iptables -A INPUT -p tcp -s 91.83.45.0/24 -d 91.83.45.77 --sport 513:65535 --dport 22 -
j ACCEPT

iptables -A OUTPUT -p tcp -s 91.83.45.77 -d 91.83.45.0/24 --sport 22 --dport
513:65535 -j ACCEPT


#ezen kívül mindent visszautasítunk
iptables -A INPUT -j DROP

iptables -A OUTPUT -j DROP
```

Ez egy nagyon alapvető konfiguráció, amennyiben más szolgáltatások is futnak a rendszeren, új sorokat is hozzá kell adnunk. Például a Novell Sentinel Log Manager webes felületének használatához engedélyeznünk kell a kommunikációt a 8443-as porton. Az iptables man oldala megtalálható a http://linux.die.net/man/8/iptables oldalon. Bármilyen hálózatra kötött rendszer esetén elengedhetetlen a tűzfal használata.

1.5. Naplózással kapcsolatos jogszabályok/szabványok

A naplózással kapcsolatos szabványok, előírások, ajánlások ismerete elengedhetetlen, ha egy naplózó rendszert tervezünk. Az informatikai rendszerek naplózásának követelményeit a 223/2009. (X.14) korm. rendelet 15. paragrafusa tartalmazza. Ez a rendelet csak az elektronikus közszolgáltatást nyújtó rendszerekre vonatkozik, egyéb informatikai rendszerekre nem. A hitelintézetekre és a pénzügyi szolgáltatásokra például az 1996. évi CXII. törvény 13/C paragrafusa vonatkozik. Hazánkban ez a törvény felel meg a bankokra vonatkozó nemzetközi megállapodásnak, a BASEL II-nek. Az említett 223/2009 X.14 kormány rendelet 15. paragrafusa, illetve az 1996-os CXII. törvény 13/C paragrafusa megtalálható a mellékletben. Az

információbiztonsággal foglalkozó szabványok általában tartalmaznak naplózással kapcsolatos követelményeket. A két leggyakrabban alkalmazott, információbiztonsággal foglalkozó dokumentum a COBIT(Control Objectives for Information and related Technologies), melyet az ISACA(Information Systems Audit and Control Association) adott ki, illetve az ISO/IEC 27002:2005 Code of practice for information security management szabvány. A COBIT jellemzően csak az elveket határozza meg, míg az ISO/IEC 27002:2005 a gyakorlati megvalósításhoz tartalmaz útmutatást, ezért a két dokumentum együttes alkalmazása ajánlott. A COBIT aktuális verziója letölthető a http://www.isaca.org/Knowledge-Center/cobit/Pages/Downloads.aspx oldalon. Érdemes még megemlíteni a HIPAA(Healt Insurance Portability and Accountability Act) szabályrendszerét, amely az egészségügyi adatok kezelésének módját határozza meg. A szabályrendszer elérhető a www.hipaa.org címen. Amikor egy rendszer tervezésénél kiválasztjuk a használni kívánt naplózó szoftvert, győződjünk meg arról, hogy milyen törvényi szabályozás vonatkozik a rendszerünkre, és a kiválasztott szoftver képes lesz-e eleget tenni ezeknek.

1.5.1. A naplózással kapcsolatos RFC dokumentumok

Az RFC (Request For Comment) valójában az internettel illetve a hálózatokkal kapcsolatos protokollok, eljárások leírását tartalmazza. Az egyes RFC dokumentumokra a számukkal hivatkozhatunk. A naplózás témaköréhez tartozó RFC dokumentumok az RFC 3164[4], az RFC 3195[5], és az RFC 5424[6].

Az RFC 3164 a BSD syslog protokoll leírását tartalmazza. Ez volt az első ajánlás, melyet a naplózó rendszerekben alkalmaztak. A legfőbb jellemzői a következők:

- UDP protokollt használ (Nem megbízható adatküldés, nem felel meg a mai követelményeknek)
- Plain text adatküldés (egyszerű szöveges adatokat továbbít a hálózaton)
- Korszerűtlen időbélyeg alkalmazása
- Nem megoldott a hitelesítés
- Nem ellenőrzött az adatok integritása

Az RFC 3195 már egy sokkal fejlettebb, megbízható továbbítást biztosító ajánlás, melynek legfőbb jellemzői a következők:

- TCP protokollt használ (nyugtázott adatküldés, így nem vesznek el a küldött bejegyzések)
- SASL authentikációt használ
- Transport Layer Security használata az átvitel során (biztonságossá teszi az adatátvitelt)

Az RFC 5424 a syslog protokoll leírását tartalmazza, melynek legfőbb jellemzői a következők:

- TCP protokollt használ (nyugtázott adatküldés, így nem vesznek el a küldött bejegyzések)
- SASL authentikációt használ
- Transport Layer Security használata az átvitel során (biztonságossá teszi az adatátvitelt)
- Javított üzenetformátum (megfelelő időpecsét, FQDN küldése)
- Támogatja a többsoros üzeneteket
- Támogatja az UTF-8 karakterkódolást

2. Novell Sentinel Log Manager

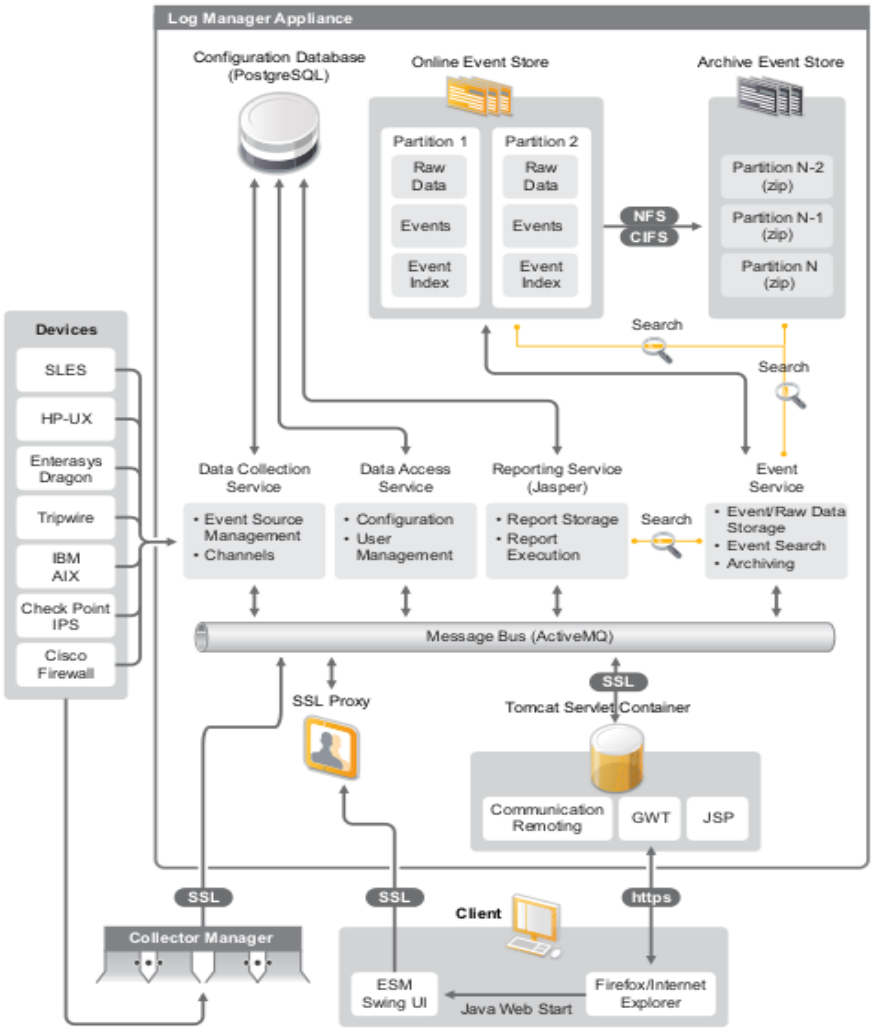
2.1. Novell Sentinel Log Manager rövid bemutatása

A Sentinel Log Manager egy komplett napló menedzsment megoldás, melyet a Novell adott ki. Főbb tulajdonságai a következők:

- naplófájlok automatikus gyűjtése
- nagy teljesítményű, valós idejű feldolgozás
- elosztott keresés tetszőleges szempontok szerint
- jelentéskészítés
- biztonságos adattovábbítás TLS/SSL segítségével
- archiválás 10:1 arányú tömörítéssel

A termék részletes specifikációja és a hozzá tartozó dokumentációk elérhetők a http://www.novell.com/hu-hu/products/sentinel-log-manager/ oldalon.

A Sentinel Log Manager felépítése a 2.1-es ábrán látható.



2.1.ábra Novell Sentinel Log Manager felépítése

(A táblázat forrása a Novell Sentinel Log Manager dokumentációja, amely a http://www.novell.com/documentation/ címen érhető el.)

2.2. Novell Sentinel Log Manager telepítése Xen virtuális környezetbe

Többféle telepítési mód közül választhatunk, ha a Novell Sentinel Log Managert szeretnénk telepíteni. Az egyik, ha egy létező Suse Linux Enterprise Server 11-es verzióra telepítjük. A másik lehetőség, ha virtuális környezetbe telepítjük. Kétféle virtuális környezet közül választhatunk, az egyik a Vmware ESX/ESXi, melyből legalább 3.5-ös, vagy 4-es verzió ajánlott, a másik a Xen 3.1.1. Az általam használt Sentinel Log Managert egy Xen virtuláis környezetbe telepítettem.

A telepítéshez a következő lépésekre volt szükség:

A http://download.novell.com webhelyről letöltöttem a Xen virtuális gépet, amely a Sentinel Log Managert tartalmazza.
Mivel a fájl tar.gz kiterjesztésű, ezért a tar -xvzf paranccsal ki kellett csomagolnom. A kicsomagolás után két fájl került a mappába, az egyik egy raw kiterjesztésű, a másik pedig egy xenconfig kiterjesztésű fájl.
A xenconfig fájl tartalmaz egy előre elkészített konfigurációt, melyet módosítani kell az aktuális rendszernek megfelelően.
Az általam használt xenconfig fájl a következőket tartalmazza:

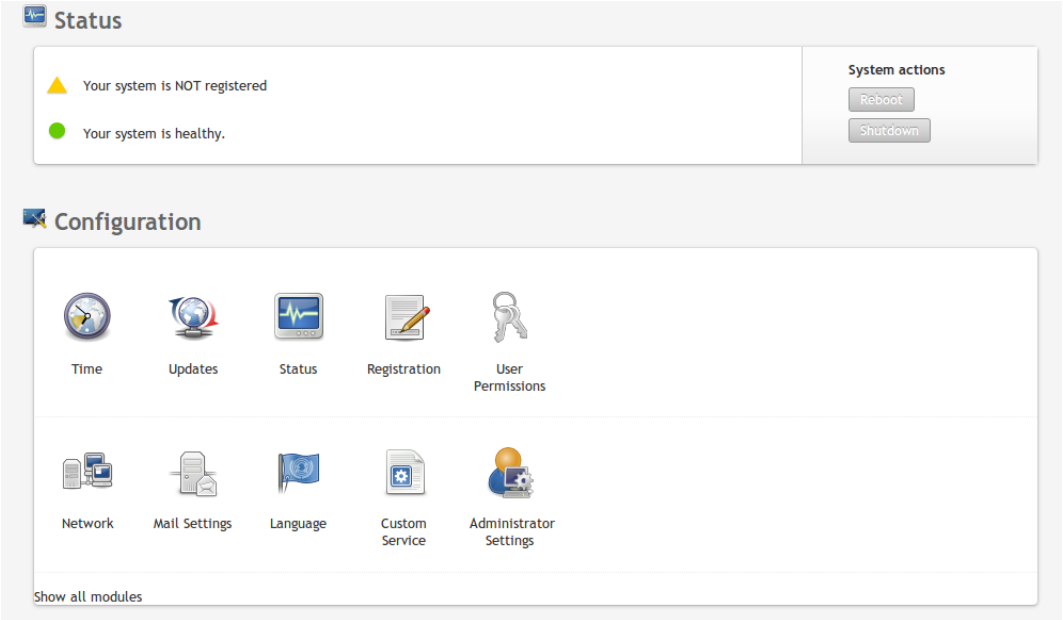
```
# -*- mode: python; -*-
name="Sentinel_Log_Manager_1.1.0.2_64_Xen"
memory=1024
disk=[
"tap:aio:/home/vipuser/logmanager/Sentinel_Log_Manager_1.1.
0.2_64_Xen-
0.783.0/Sentinel_Log_Manager_1.1.0.2_64_Xen.x86_64-
0.783.0.raw,xvda,w" ]
vif=[ "bridge=br0" ]
vcpus = 4
```

A name bejegyzés a virtuális gép nevét tartalmazza. A memory bejegyzés a gép számára engedélyezett fizikai memória méretét határozza meg, ami ebben az esetben 1024 MB. A disk bejegyzés a virtuális gép raw fájljának helyét mutatja. A vif bejegyzés a hálózati konfigurációt határozza meg. A vcpus bejegyzés arra utal, hogy az adott gép 4 magot használhat a rendszeren.
Ezután ki kell adni az `xm create sentinel.xenconfig` parancsot, hogy létrejöjjön a virtuális gép.
Azt, hogy az előző művelet sikeresen lezajlott, az `xm list` paranccsal ellenőrizhetjük, ez kilistázza a virtuális gépeket.
A telepítés elkezdéséhez adjuk ki az
`xm console Sentinel_Log_Manager_1.1.0.2_64_Xen` parancsot.

Ezután elindul a Sentinel Log Manager Beállítása parancssoros felületen keresztül. Itt csak az alap beállításokat kell megadnunk, melyek a rendszer működéséhez szükségesek. Ilyen beállítások a hosztnév, a hálózat beállításai, dátum és idő beállítások, és a jelszavak megadása, amelyekkel később hozzáférhetünk a rendszerhez.

2.3. Sentinel Log Manager, hozzáférés telepítés után

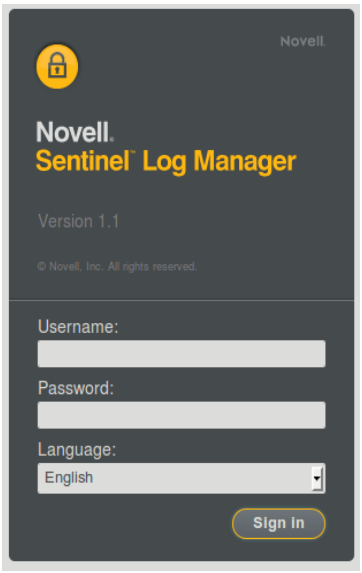
Miután a programot telepítettük, webes felületen menedzselhetjük a legtöbb funkciót. Lehetőségünk van elérni a Webyast-ot, ahol a gépre vonatkozó alapvető beállításokat módosíthatjuk, mint például az idő, nyelvi beállítások, hálózati és levelezőszerver beállítások, frissítések, felhasználói jogosultságok. Gyakorlatilag minden fontos beállítást elvégezhetünk ezen a grafikus felületen. A WebYast eléréséhez a böngészőnkbe írjuk be a szerverünk IP címét, használjunk https protokollt és az 54984-es portot. A Webyast kezdőlapjáról készült képernyőkép a 2.2-es ábrán látható.
A fő webes felület, melyen a naplózó rendszert menedzselhetjük a 8443-as porton érhető el alapértelmezetten, szintén https használatával.



2.2 ábra Webyast kezdőlapja

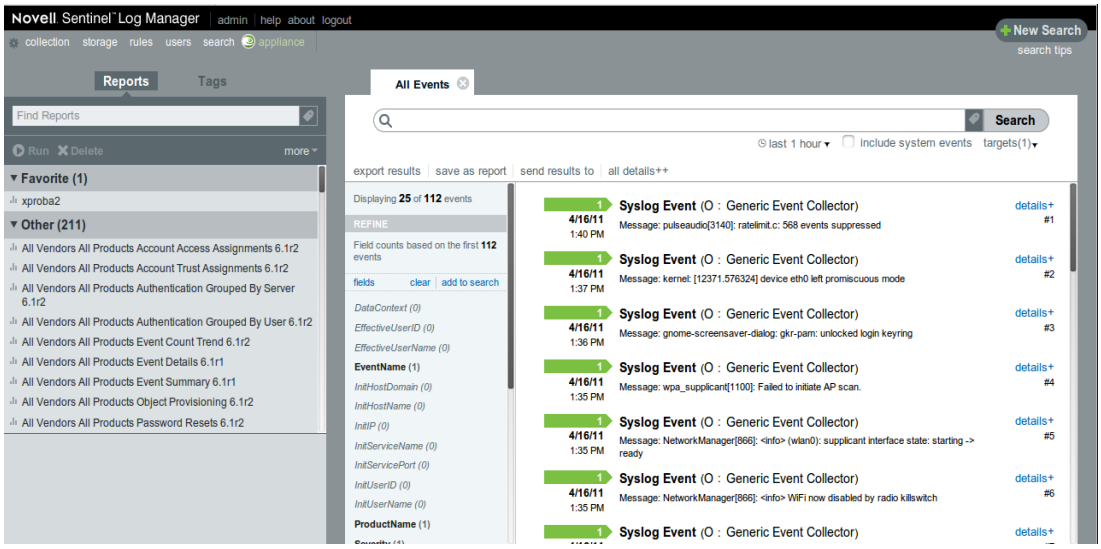
2.4. Sentinel Log Manager felületének áttekintése, és alapvető beállítások

A webes kezelőfelületet a 8443-as porton érhetjük el, https protokoll használatával. Miután beütöttük a címet a böngészőbe, egy bejelentkező oldal fogad minket(2.3-as ábra), ami a sikeres authentikáció után átirányít a Sentinel kezelőfelületére.



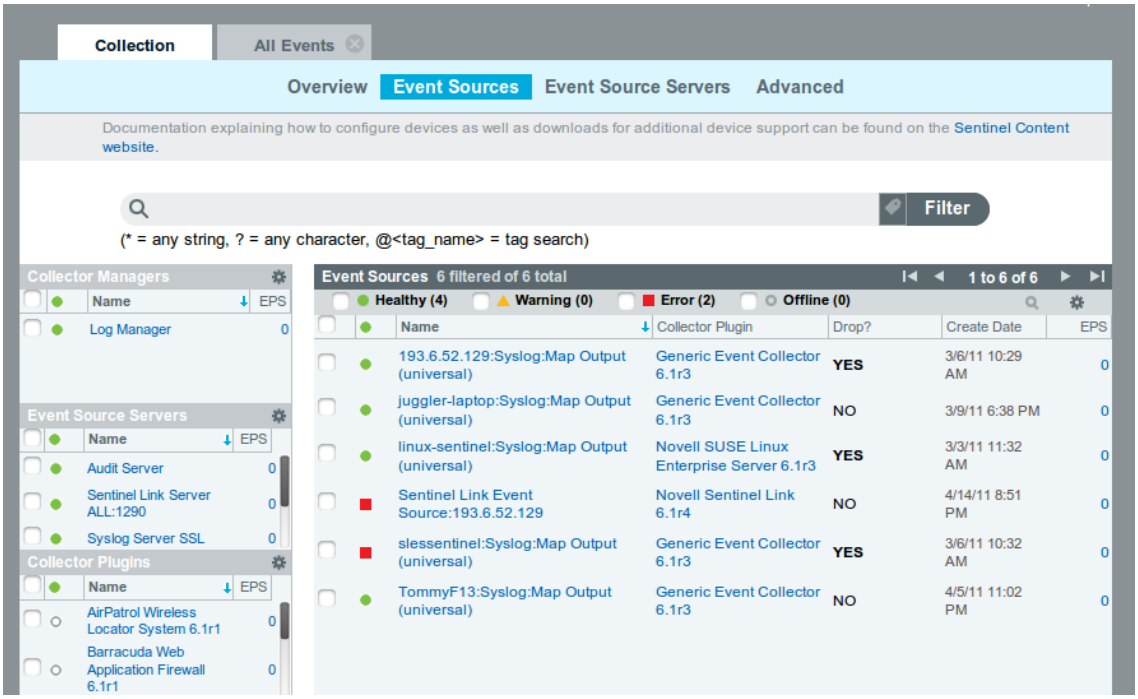
2.3. ábra Novell Sentinel Log Manager webes felületének bejelentkező ablaka

A megjelenő felület nagyon jól strukturált és átlátható, ahogy az a 2.4-es ábán látható. A kezdőlapon az utolsó egy óra naplóbejegyzéseit láthatjuk, illetve a keresés, szűrés és a jelentéskészítés opciókat.



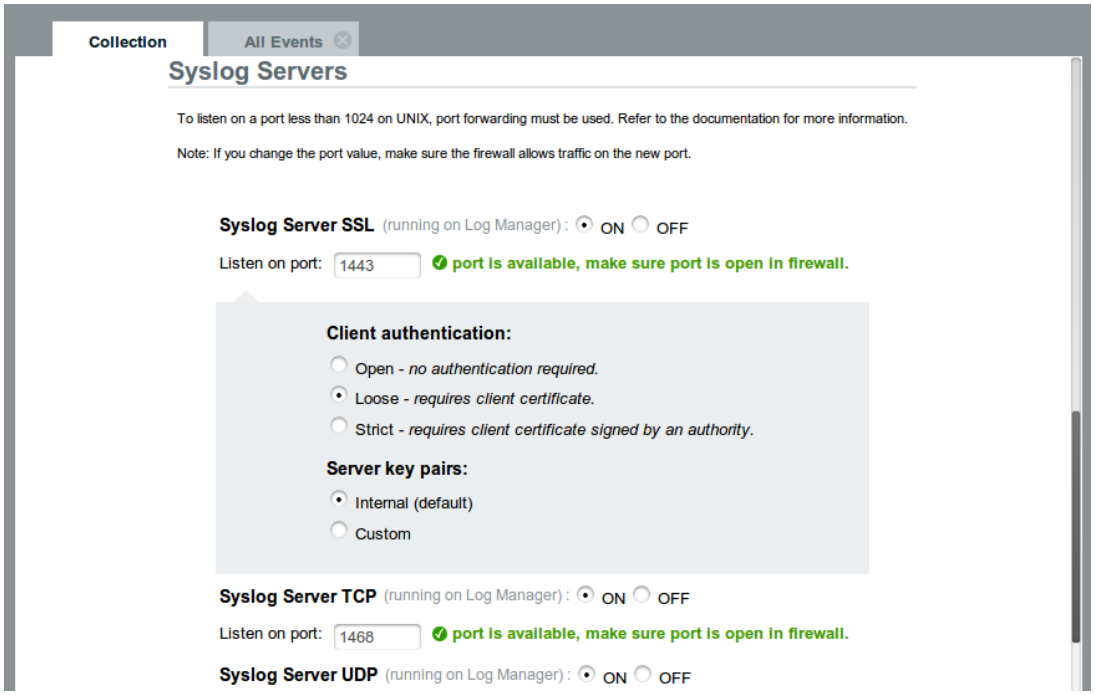
2.4.ábra Novell Sentinel Log Manager kezdőlapja

A collection menüpontra kattintva láthatjuk, hogy az elmúlt egy percben átlagosan hány bejegyzést fogadott a szerver. Itt áttekinthetjük az összes rendszert, amelyektől a szerver naplóbejegyzéseket fogad, és látható ezek állapota is(2.5-ös ábra).



2.5. ábra Novell Sentinel Log Manager források áttekintése

Az Event source servers fülön beállíthatjuk, hogy milyen portokon fussanak a különböző logszerverek, melyek az üzeneteket fogadják, illetve, hogy milyen hitelesítést alkalmazzanak, ahogy az a 2.6-os ábrán látható. Itt engedélyezni kell a „syslog server tcp”, illetve a „syslog server ssl” folyamatokat. A TCP syslog szerver az 1468-as porton, míg az SSL-t használó syslog szerver az 1443-as porton hallgatózik. Az SSL-nél háromféle beállítást alkalmazhatunk a tanusítványokra vonatkozóan, melyek a következők: nem szükséges tanusítvány, érvényes X.509 kliens tanusítvány szükséges, megerősített X.509 tanusítvány szükséges. Én a második beállítást fogom alkalmazni. Az üzenetek fogadása authentikáció nélkül nem javasolt, mivel így bárki küldhet üzeneteket a szerverre, aki ismeri az IP címet és a portot, amelyeken a szerver fut. Így könnyen indítható szolgáltatásmegtagadás (DoS) típusú támadás a szerver ellen úgy, hogy nagy mennyiségű naplóbejegyzést küldünk rövid idő alatt a szerver felé, így a rendszer nem tudja fogadni az üzeneteket a saját rendszerünk gépeiről.



2.6. ábra Novell Sentinel Log Manager syslog szerverek beállítása

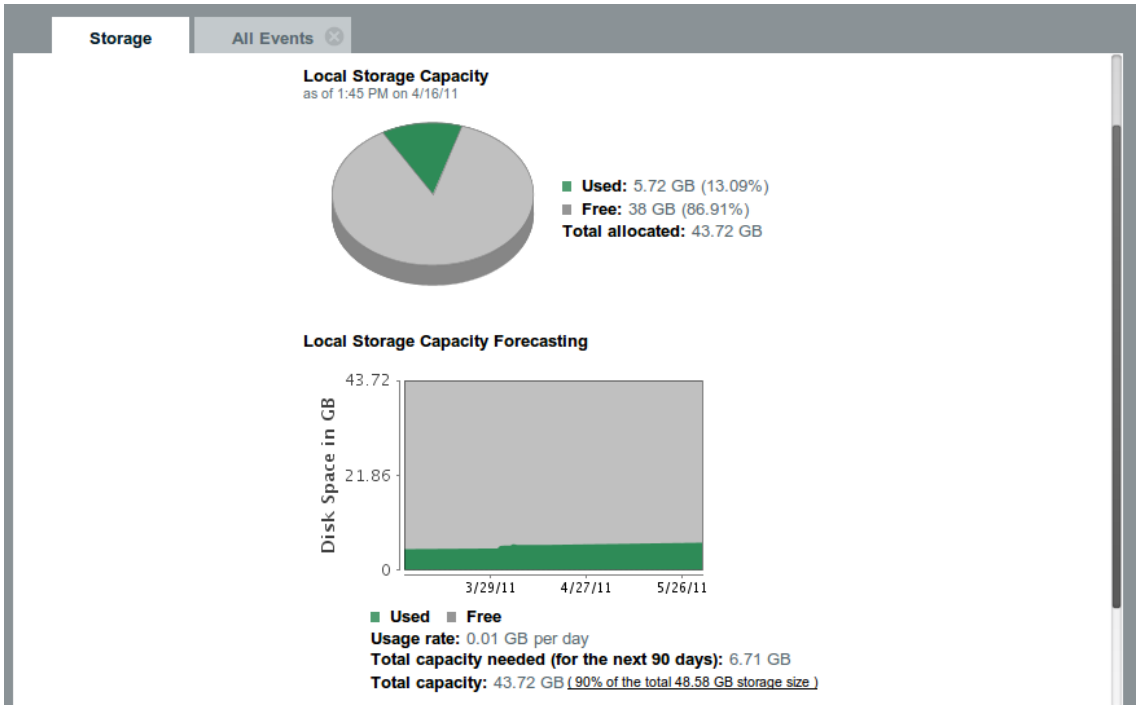
A storage menüpontra kattintva a naplótárolással kapcsolatos információkat és beállításokat látjuk. Ezen a lapon egy áttekintő képet kapunk a rendszer pillanatnyi tárhely foglaltságáról, láthatunk egy előrejelzést a következő 90 nap tárhelyfoglalásáról,

melyet az eddigi átlag alapján számol a rendszer (2.7-es ábra). Lehetőség van hálózati meghajtókat használni a tároláshoz, ezeket szintén a storage lapon állíthatjuk be. A különböző rendszerek tárhely szükséglete függ az üzenetek számától, és attól, hogy meddig kell megőriznünk a naplófájlokat, ezért nehéz előre megbecsülni, hogy mekkora tárhelyre lesz szükség a működéshez. A következő képlet segítségével megbecsülhetjük, hogy mennyi tárhelyre lesz szükségünk a szerveren.

*(Átlagos bejegyzés méret byte-ban) * (napok száma amíg az adatot tároljuk) * (bejegyzések másodpercenként) * 0,00007 = A szükséges lokális tárhely mérete (GB)*

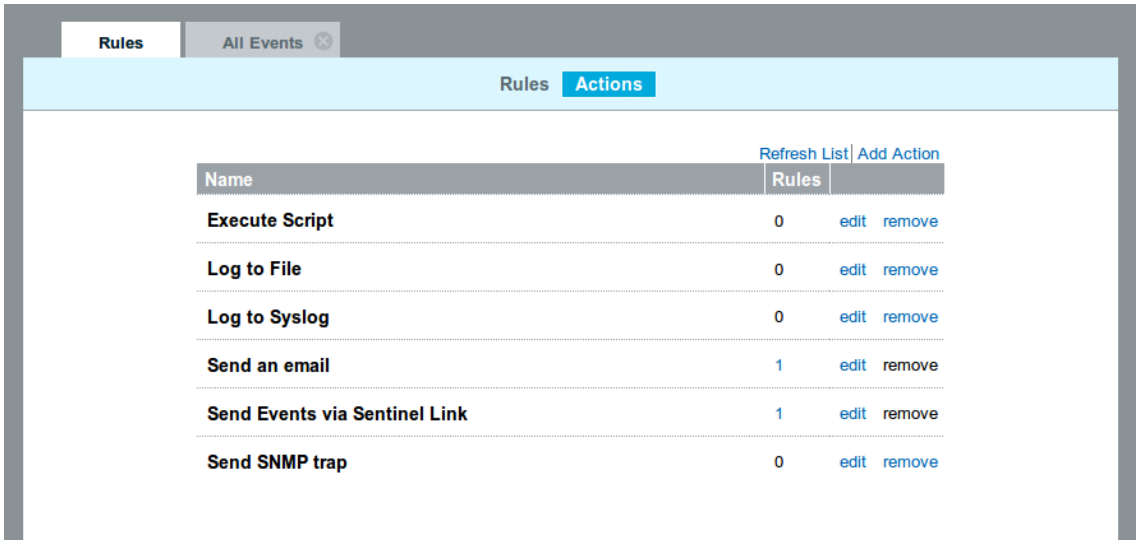
Amikor a lokális tárhely foglaltsága egy bizonyos szint alá csökken, a Sentinel Log Manager betömöríti az adatokat, és áthelyezi őket egy hálózati tárolóra. A hálózati tároló szükséges méretét a következő képlet segítségével becsülhetjük meg:

*(Átlagos bejegyzés méret byte-ban) * (napok száma amíg az adatot tároljuk) * (bejegyzések másodpercenként) * 0,00002 = A szükséges hálózati tárhely mérete (GB)*



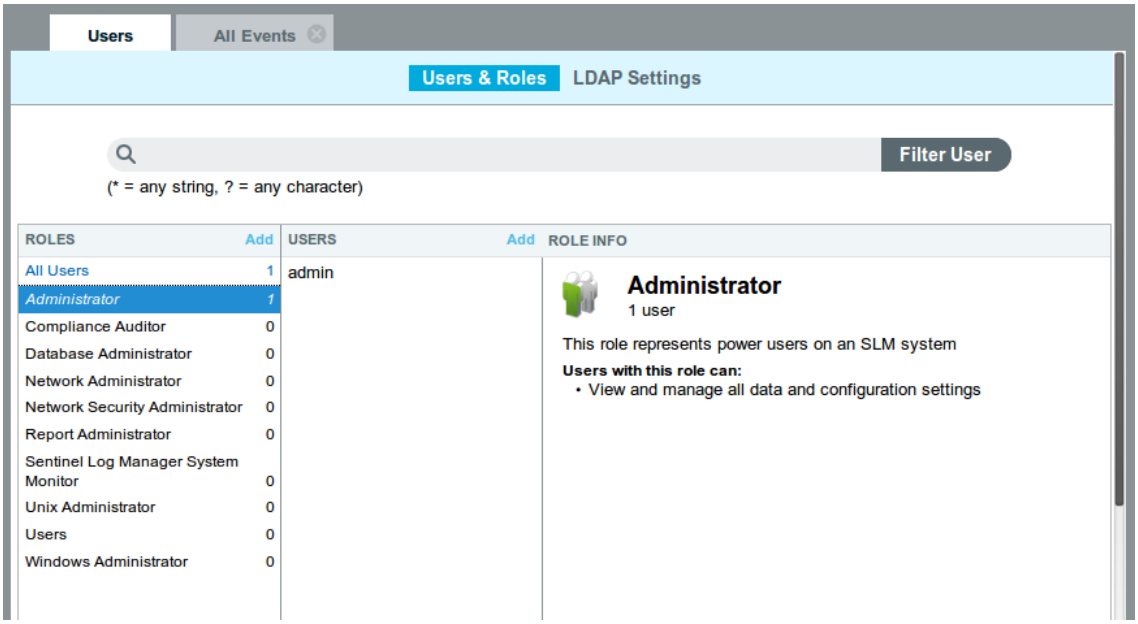
2.7. ábra Novell Sentinel Log Manager tárhely foglaltság

A rules menüpontra kattintva egyedi szabályokat hozhatunk létre(2.8-as ábra). Beállíthatjuk, hogy bizonyos esemény bekövetkezésekor futtasson egy scriptet, küldjön e-mailt, vagy akár küldjön egy SNMP trap-et.



2.8. ábra Novell Sentinel Log Manager szabályok beállítása

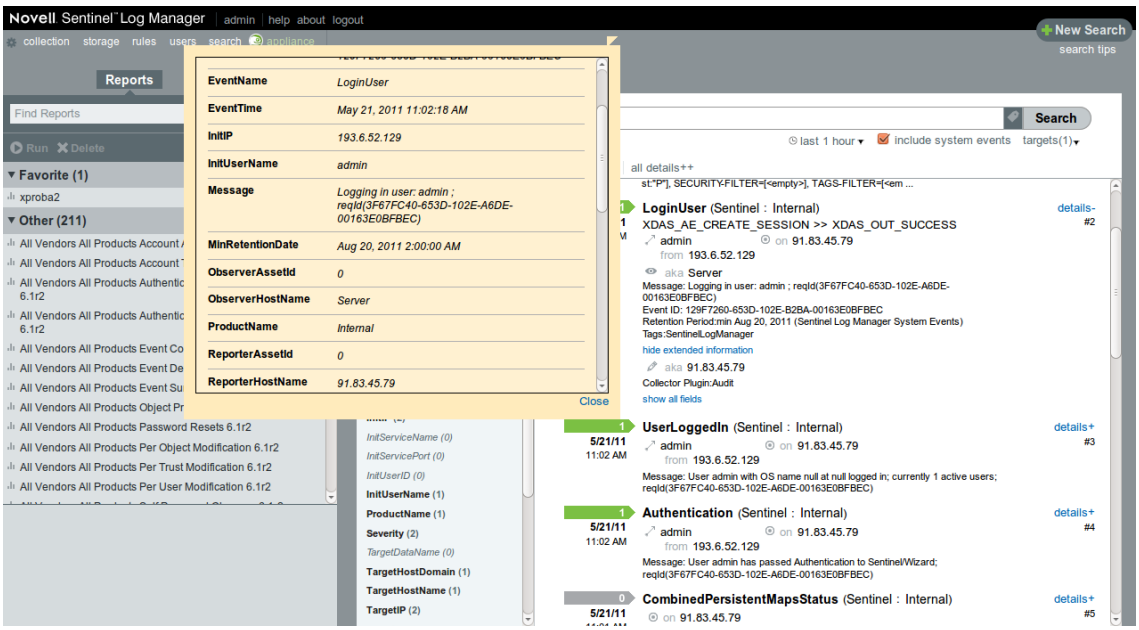
A users menüpontra kattintva új felhasználókat adhatunk hozzá, és megadhatjuk a hozzájuk tartozó szerepkört, illetve jogosultságokat, ahogy az a 2.9-es ábrán látható. Amennyiben rendelkezünk LDAP szerverrel, megadhatjuk a szerver paramétereit, így a Sentinel az LDAP szerverről kéri le a felhasználói adatokat az authentikációhoz.



2.9. ábra Novell Sentinel Log Manager felhasználók kezelése

A naplózás során nagy mennyiségű adatot gyűjtünk össze, melyek a rendszerünk aktuális, és korábbi állapotát mutatják. A Sentinel Log Manager segítségével egyszerűen szűrhetünk, kereshetünk az adatok között, így tökéletes képet kaphatunk a rendszerünk bizonyos részeiről. Tegyük fel, hogy szeretném látni az elmúlt 30 napban történt SSH belépések bejegyzéseit. Ehhez nem kell mást tennem, mint kiválasztani a search menüpontot, beírni a keresés mezőbe, hogy „sshd user authenticated”, és kiválasztani, hogy az utolsó 30 nap bejegyzéseit mutassa a rendszer. A kapott eredményhalmaz szűkíthető a bejegyzésektől balra található részen lévő mezők segítségével, így megkaphatjuk például az egy adott IP címről történő belépéseket, de gyakorlatilag bármilyen rendelkezésre álló információ alapján szűkíthetjük az eredményhalmazt. A keresés során használhatjuk a logikai AND, OR és NOT operátorokat, de ügyeljünk rá, hogy végig nagybetűvel írjuk őket, hogy a rendszer

helyesen értelmezze. Az előző példát kiegészítve, tegyük fel, hogy nem a felhasználókkal kapcsolatos SSH eseményekre vagyok kíváncsi, hanem az egyéb bejegyzésekre, például a szolgáltatás indulására vagy leállására, de nem tudom, hogy pontosan milyen bejegyzés kerül ilyenkor a naplófájlba. Ha egyszerűen az „ssh” kifejezést írom be, az eredményhalmaz túl nagy lesz, azonban az „ssh AND NOT user” kifejezésre kapott eredményhalmazban megtalálom a keresett „sshd service start” és „sshd service terminated” bejegyzéseket. Az egyes bejegyzésekre kattintva megtekinthetjük a hozzájuk tartozó részletes információkat, ahogy a 2.10-es ábra mutatja.



2.10. ábra Novell Sentinel Log Manager keresés szűkítése, részletes információk

A kereséseinket elmenthetjük jelentésként, így később nem kell újra elkészítenünk őket, csak egyszerűen kiválasztani a bal oldalt található listából és lefuttatni. A jelentések listában számos előre elkészített jelentés található. Nem csak egyszerű lista szerű jelentések készíthetők. Van lehetőségünk összegző jelentések készítésére, amelyek diagramokon szemléltetik a kapott eredményeket. A jelentést ütemezhetjük, és elküldhetjük magunknak e-mailben előre megadott időközönként, így a mindennapos feladatok egyszerűen automatizálhatók.

2.5. SLES 11 SP1 kliens beállítása

A tesztkörnyezetemhez először egy Suse Linux Enterprise Server 11 SP1-es verziót adtam hozzá. Ez a linux disztribúció alapértelmezetten Syslog-ng-t használ. A Novell Sentinel Log Manager nagyon sok más mellett a Syslog-ng-vel is kompatibilis, képes fogadni, tárolni, és feldolgozni az ilyen forrásból érkező naplóbejegyzéseket. Három féle protokollon tudunk kapcsolódni a syslog szerverhez, ha syslog-ng-t használunk. Ezek a protokollok az UDP, a TCP és az SSL-el kiegészített TCP. A következő példában a TCP protokoll beállítását fogom bemutatni. Az SSL titkosított adatfolyamot használ, ezért mindenképpen ezt célszerű beállítani egy éles rendszeren, ahol fontos az adatok biztonsága.

Ahhoz, hogy a rendszerünk a naplóbejegyzéseket elküldje a Sentinel Log Manager syslog szerverének, módosítanunk kell a syslog-ng konfigurációs fájlt. Az én rendszeremen ez a konfigurációs fájl a /etc/syslog-ng/syslog-ng.conf. Mielőtt módosítanánk a fájlt, készítsünk róla biztonsági másolatot. Ezt úgy tehetjük meg a legegyszerűbben, ha a könyvtárban állva kiadjuk a következő parancsot:

cp syslog-ng.conf{,.original}

A fájlban azt kell beállítanunk, hogy minden bejegyzést továbbítson a szerverünk felé, melyet a következő sorok felvételével tehetünk meg:

```
destination logserver { tcp("91.83.45.79"
                          port(1468)); };
log { source(src);
      destination(logserver); };
```

Ezután újra kell indítanunk a syslog-ng-t, hogy az új beállításokat betöltse a rendszer.

Az újraindítást a /etc/init.d/syslog restart paranccsal tehetjük meg.

A konfiguráció teszteléséhez adjuk ki a következő parancsot:

logger test message

Ez a parancs egy naplóbejegyzést készít, így nem kell várnunk a következő rendszereseményre, hogy leteszteljük a kapcsolatot a szerverrel.

2.6. Rsyslog kliens beállítása

A syslog-ng mellett a másik naplózó szoftver, amelyet számos linux disztribúcióban megtalálhatunk, az rsyslog. Amennyiben nem vagyunk biztosak benne, hogy az általunk használt disztribúció milyen naplózó szoftvert használ, használjuk a következő parancsot:

ps aux | grep syslog

Így megkapjuk az éppen futó syslog folyamatot. Amennyiben rsyslogd szerepel a parancs kimenetében, a következő rövid leírás segít beállítani az üzenetek továbbítását. Először is keressük meg az rsyslog config fájlt. Az én rendszeremen ez a /etc/rsyslog.conf elérési útvonalon található. Első lépésben készítsünk róla egy biztonsági másolatot mielőtt módosítani kezdenénk. A másolat készítéséhez adjuk ki a kövezkező parancsot a könyvtárban állva:

cp rsyslog.conf{,.original}

Most már nyugodtan módosíthatjuk a fájlt, hiszen ha valamit elrontunk, egyszerűen visszamásoljuk az eredetit, és kezdhetjük elölről.

A konfigurációs fájlhoz adjuk hozzá a következő sort:

. @@91:83.45.79:1468

Ezzel beállítottuk, hogy az összes bejegyzés továbbításra kerüljön, a megadott IP címen és porton hallgatózó syslog szerverhez. A *.* az összes bejegyzést jelzi, a két @ karakter pedig a tcp protokoll beállításához szükséges. A beállítások érvénybe lépéséhez újra kell indítanunk a folyamatot. Ehhez adjuk ki a következő parancsot:

/etc/init.d/rsyslog restart

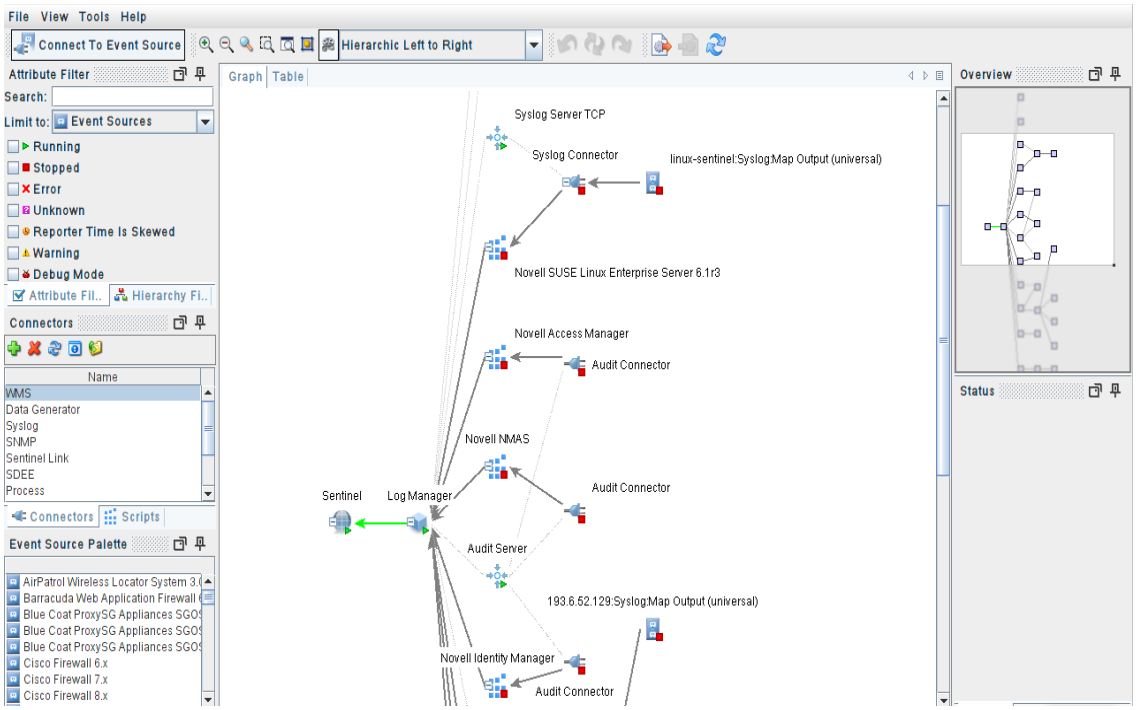
Az újraindítás után adjuk ki a következő parancsot a beállítás teszteléséhez:

logger testrsyslogd

Ezzel egy naplóüzenetet küldünk a szerver felé, így nem kell megvárnunk a következő eseményt, hogy ellenőrizni tudjuk a kapcsolatot.

2.7. Nem syslog alapú kliens beállítása

A Novell Sentinel Log Manager egyik legnagyobb előnye, hogy nem csak syslog alapú kliensek felügyelhetők vele, így egy komplex informatikai rendszer is egyetlen helyről felügyelhetővé válik. A legnépszerűbb alkalmazásokhoz és eszközökhöz előre elkészített adatgyűjtők állnak rendelkezésre. Az összes kliens egyszerűen menedzselhető az Event Source Manager segítségével, melyet a Novell Sentinel Log Manager webes felületéről indíthatunk. Az Event Source Manager indítása után megjelenik egy hierarchikus felépítésű áttekintő, amely az egész rendszert ábrázolja (2.11. ábra). Itt egyszerűen adhatunk hozzá új elemeket a felületen található listákból. Új adatgyűjtő hozzáadásakor egy varázsló segítségével adhatjuk meg a szükséges beállításokat, így használata egyszerű.



2.11. ábra Novell Sentinel Log Manager event source manager felülete

2.8. Hardverkövetelmények és tesztelés

A Novell háromféle hardverkövetelményt határoz meg, a rögzített események számától függően. 500 esemény rögzítéséhez másodpercenként a Novell legalább egy 4 magos Intel Xeon E5450-es processzort, 4 GB memóriát, és két darab 500 GB-os merevlemezt ajánl, melyeknek fordulatszáma legalább 7200 RPM, összekötve egy hardveres RAID 1-es tömbben, 256 MB cache-el. A maximális, 7500 esemény/másodperc rögzítéséhez az ajánlott hardver két darab Intel Xeon X5470(összesen 8 mag), 8 GB memória, 16 darab 600 GB-os, 15000 RPM-es merevlemez, hardveres RAID 10 tömbben, 512 MB gyorsítótárral.

Nagyon nehéz előzetesen megbecsülni, hogy a rendszerünk hány eseményt fog generálni másodpercenként. Mindenképpen javasolt az átlagos terheléshez viszonyítva túlméretezni a naplózó rendszert, így ha esetleg később bővítjük a rendszert, vagy olyan hiba keletkezik, amely nagy mennyiségű naplóüzenetet generál, a rendszerünk stabil marad. A Novell kiadott egy ingyenes verziót is a Sentinel Log Managerből, melynek az egyetlen korlátja, hogy másodpercenként maximum 25 üzenetet tud fogadni. Ez a verzió biztosan nem megfelelő, ha egy komplett rendszer napló menedzsment feladatait szeretnénk ellátni, amely tartalmaz hálózati eszközöket, adatbázis szervereket, behatolás érzékelőket, vírusírtókat, webszervereket, és egyéb eszközöket, azonban teszteléshez ideális, mivel a fizetős szoftver teljes funkcionalitását kínálja, időkorlát nélkül.

Az általam készített tesztkörnyezet a Sentinel minimális hardverkövetelményeinek felel meg. A következő részben megvizsgálom, hogy ez a minimális hardver milyen teljesítményre képes, hány eseményt képes rögzíteni úgy, hogy a rendszer használható marad. A teszteléshez készítettem egy bash scriptet, amely megadott időközönként tesztüzeneteket küld a szerver felé. Az üzenetküldést a következő egyszerű script végzi:

```
#!/bin/bash
i=0;
while true; do
        logger  "testmessage $i";
        i=$((i+1));
        sleep 0.1;
done
```

A scriptben a logger parancsot használtam, melynek segítségével egyedi üzenetek küldhetők a syslog számára. A sleep után található számmal szabályozható az üzenetküldés gyakorisága. A példában látható érték(0,1) esetén átlagosan 10 üzenetet küld másodpercenként.

A 2.12-es ábrán látható táblázatban összefoglaltam a teszt eredményeit. A „Sleep idő” oszlopban az az érték látható, amelyet a script-ben beállítottam a sleep parancs paramétereként. Amikor a script futása ehhez a parancshoz ér, ennyi másodpercig várakozik. Ez azért szükséges, hogy szabályozni lehessen az üzenetek mennyiségét. Ahol a „Nincs korlát” szöveg szerepel az oszlopban, ott kivettem a sleep parancsot, így a kliens gép teljesítményének megfelelő maximális naplóbejegyzést küldi a szervernek. A „Kliensek száma” oszlop a szerverre csatlakoztatott kliensek darabszámát mutatja. Az első sorban ez az érték nulla, így látható a szerver kihasználtsága, amikor alapállapotban van. Az egyik legfontosabb oszlop a „Load a szerveren”. A load érték linuxos környezetben a terhelés átlagát mutatja. "Azt jelzi, hogy hány folyamat várakozik a processzor feldolgozási sorában, vagyis a futó vagy feldolgozásra váró folyamatok számát. (…) Minél magasabb az átlag, annál valószínűbb, hogy a gép szenvedni kezd a túlterhelés alatt, és annál valószínűbb, hogy a felhasználók elkezdik a mellékünket tárcsázni."[1] Az utolsó, 7,1-es mért értéknél a webes felület érezhetően lelassult, és nehezen használhatóvá vált, így hosszú távon nem megfelelő, ha ilyen terhelést kap a szerver, azonban a naplóüzeneteket így is gond nélkül tudta fogadni. Az „EPS érték a szerveren” a másodpercenként érkező naplóbejegyzések számát mutatja, az angol Event Per Second rövidítése. Látható, hogy a szerver által mért érték a második, harmadik, és ötödik sorban eltér a várt értéktől. Mivel a sleep parancs paramétere 0,1, illetve 0,01, ezért egy gép esetén az üzenetek száma másodpercenként 10 és 100, két gép esetén 0,01-os sleep idővel pedig 200 kellene, hogy legyen. A két érték azért eltérő, mivel a script futási ideje megnöveli a két küldött üzenet között eltelt időt, ami a sleep paraméterének csökkentésével egyre jobban érezhetővé válik. A CPU terhelés folyamatosan ugrált az üzenetek fogadása közben, ezért a mért adatok tájékoztató jellegűek. A valódi terhelést a load érték mutatja a legjobban. A „Szabad mem.” oszlopban a szabad fizikai memória mérete látható megabájtban mérve. A rendszerben összesen 1 GB fizikai memória van, a minimális rendszerkövetelmény 0,9 GB-ot ír elő.

Jól látható, hogy a memória kihasználtsága nem növekedett a fogadott bejegyzések számával párhuzamosan.

Sleep idő (s)	Kliensek száma	Load a szerveren	EPS érték a szerveren	Szerver CPU terhelés (%)	Szabad mem.(MB)
-	0	0,3	0	0	15 MB
0,1	1	0,3	9	7,00%	18 MB
0,01	1	0,9	63	10,00%	17 MB
Nincs korlát	1	4	540	32,00%	17 MB
0,01	2	1,4	133	24,00%	15 MB
Nincs korlát	2	7,1	710	50,00%	10 MB

2.12. ábra Novell Sentinel Log Maneger teszt eredmények

2.8.1. Teszt eredmények összegzése

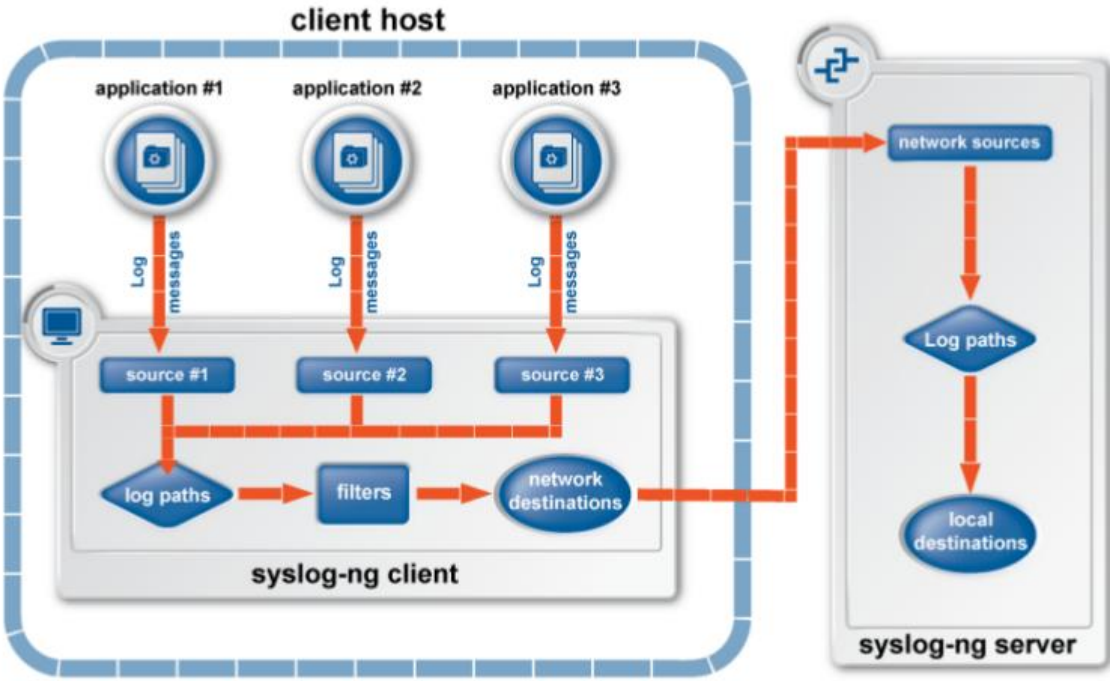
A rendszer nem rendelkezik azzal a hardver környezettel, amelyet a novell az 500 EPS terheléshez ajánl, azonban jól láthatóan a minimális követelmény is ki tud szolgálni egy ilyen számú eseményt generáló rendszert. Még akkor sem válik használhatatlanná, ha ez az érték közel másfélszer akkora lesz.

3. Syslog-ng

3.1. Syslog-ng rövid bemutatása

A Syslog-ng a Balabit Kft. terméke, 1997 óta fejlesztik, jelenleg a legújabb a 4.0-s verzió. A termékcsaládhoz három termék tartozik. Az egyik a Syslog-ng Open Source Edition, amely nyílt forráskódú, ingyenesen használható, a másik a Syslog-ng Premium Edition, amely nagyobb tudású mint az OSE változat, azonban ez már nem ingyenes, a

harmadig pedig a Syslog-ng Storebox Edition, amely komplett hardver és szoftver



megoldás. A Syslog-ng működési vázlata a 3.1-es ábrán látható.

3.1 ábra Syslog-ng működési vázlata

(Az ábra forrása a Balabit Kft. által kiadott dokumentáció, amely a következő címen érhető el: http://www.balabit.com/hu/network-security/syslog-ng/central-syslog-server/support/documentation)

3.2. Syslog-ng három változatának összehasonlítása

Az Open Source változat legnagyobb előnye, hogy ingyenes. A Sentinel Log Manager ingyenes változatával szemben, itt nincs korlátozás az eseményszámra vonatkozóan, azonban nem kapunk teljes funkcionalitást. A három változat legfontosabb tulajdonságait a 3.2-es ábrán látható táblázat foglalja össze.

Tulajdonságok	syslog-ng OSE	syslog-ng PE	syslog-ng Store Box
Üzenetek megbízható továbbítása (TCP)	✔	✔	✔
Üzenetek szűrése tartalom alapján	✔	✔	✔
Adatbázis táblák, könyvtárak, fájlok dinamikus létrehozása makrók segítségével	✔	✔	✔
IPv6 támogatás	✔	✔	-
Naplózás közvetlenül adatbázisba	✔	✔	✔
Üzenetek titkosított továbbítása (TLS)	✔	✔	✔
A legfrissebb IETF-syslog szabványok támogatása	✔	✔	✔
Üzenetek feldolgozása és módosítása	✔	✔	✔
Titkosított, aláírt, időpecséttel ellátott tárolás	-	✔	✔
Üzenetek pufferelése merevlemezre	-	✔	✔
Üzenetküldés sebességének szabályozása	✔	✔	✔
Windows támogatás	-	✔	✔
Hardver berendezés	-	-	✔
Web alapú konfigurációs felület	-	-	✔
Magas rendelkezésre állás támogatása	-	-	✔
Integrált naplókereső, böngésző és jelentéskészítő felület	-	-	✔

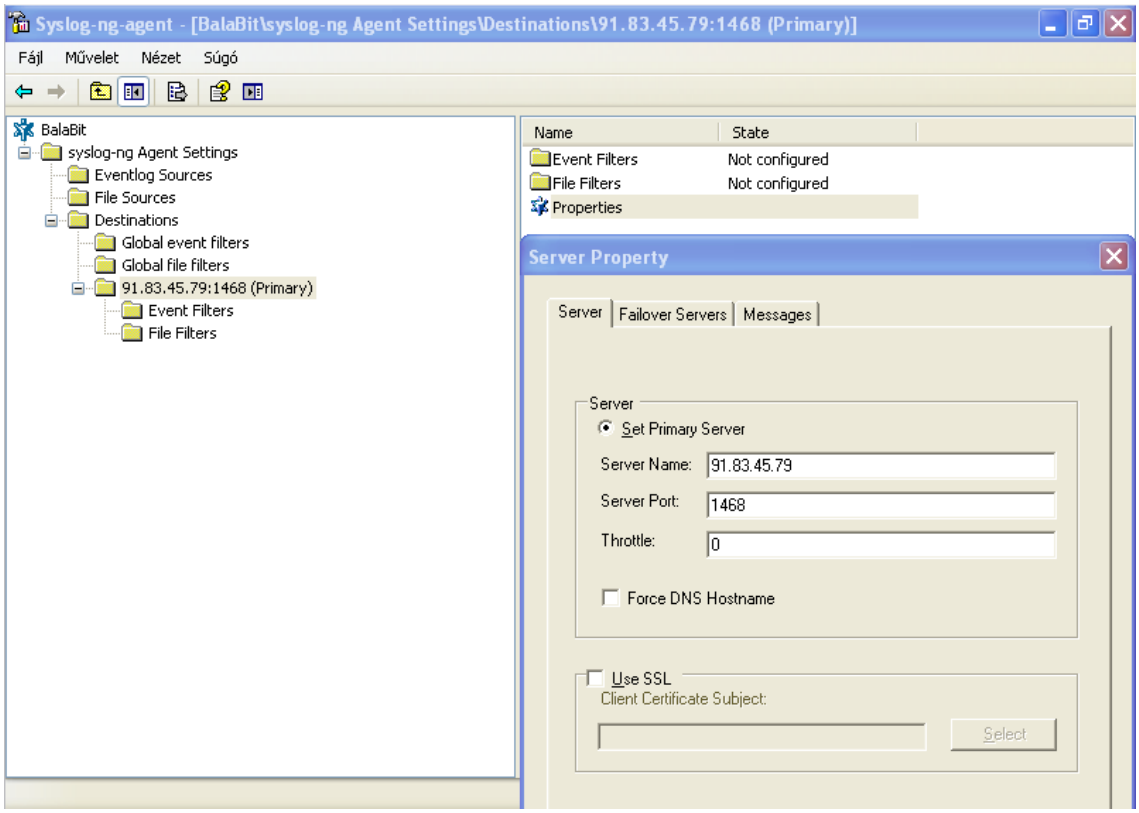
3.2 ábra Syslog-ng változatainak legfontosabb jellemzői

(A táblázat forrása a Balabit Kft. hivatalos weboldalán található összehasonlító táblázat, amely a http://www.balabit.com/hu/network-security/syslog-ng/central-syslog-server/specifications/comparison címen érhető el)

A Syslog-ng három változatának funkcióit részletesen bemutató táblázat a mellékletben található.

3.3. Syslog-ng Windows XP kliens beállítása

A Syslog-ng Windows XP alapú kliensének telepítése nagyon egyszerű, az indítás után grafikus felületen, lépésről lépésre adhatjuk meg a minimális beállításokat. A telepítés előtt meg kell győződnünk róla, hogy feltelepítettük legalább a 2.0-s .Net Framework-öt. Kétféle beállítási módszer között választhatunk. Az egyik az MMC alapú, a másik pedig XML alapú. Én az MMC alapú verziót telepítettem. A szükséges beállítások megegyeznek a syslog-ng többi verziójával, csak itt nem config fájlokat kell módosítani, hanem az MMC ablakban kell bejegyzéseket hozzáadni a hierarchikus struktúrában, ahogy az a 3.3-as ábrán látható.



3.3 ábra Syslog-ng MMC alapú konfiguráció

A kliensen nem állítottam be az üzenetek szűrését, az egyetlen beállítás amit megadtam, a cél szerver IP címe, és a használt port. A syslog-ng filter beállításaira később térek ki, mivel ez nem kliens specifikus. A klienst ezután a start menüben található parancsikonokkal tudjuk leállítani, elindítani, illetve megnyitni a hozzá tartozó MMC konfigurációs ablakot.

3.4. Syslog-ng SLES 11 kliens beállítása

A Suse Linux Enterprise Server 11 alapértelmezetten a Syslog-ng OSE verzióját használja. A letöltés után a telepítéshez adjuk ki a következő parancsot:
zypper install syslog-ng-pe-4.0.rpm
A telepítés végén az ellenőrzéshez adjuk ki a következő parancsot:
tail -f /var/log/messages
A parancs kimenetében a következő sorokat találjuk:
linux syslog-ng[1744]: syslog-ng shutting down; version='2.0.9'
linux syslog-ng[4175]: syslog-ng starting up; version='4.0.1', cfg-fingerprint='a1513496a0fc5872a43f731461b89592564f659d', cfg-nonce-ndx='0', cfg-signature='c3cf954f390d1d8383223760567052219 87abc39'

A bejegyzésekből jól látható, hogy a syslog-ng 2.09-es verziója helyett, az újonnan telepített 4.0.1-es verzió sikeresen elindult. A cfg-fingerprint és a cfg-signature két darab hash érték csak az újonnan telepített verzióban jelenik meg. Minden betöltéskor, a syslog-ng megvizsgálja, hogy a konfigurációs fájl változott-e az előző betöltés óta, és amennyiben változás történt, a központi log szerver felé jelzi. Az újonnan telepített syslog-ng konfigurációs fájlt a /opt/syslog-ng/etc/syslog-ng.conf elérési úton találjuk. Az alap beállítást, mely szerint minden üzenetet egy központi log szerver felé továbbítunk, itt is ugyanúgy kell megadni, mint az OSE változatban. Adjuk hozzá a következő sorokat a konfigurációhoz:

```
destination logserver { tcp("91.83.45.79"
                        port(1468)); };
```

```
log { source(src);
        destination(logserver); };
```

A fájl mentése után adjuk ki a /etc/init.d/syslog-ng restart parancsot. A konfiguráció tesztelését itt is a logger parancs segítségével tehetjük meg.

3.5. Syslog-ng szerver beállítása

A syslog-ng beállításait a konfigurációs fájlok szerkesztésével módosíthatjuk. Az én rendszeremen ez a fájl a /opt/syslog-ng/etc/syslog-ng.conf. Ezt a konfigurációs fájlt tölti be a syslog-ng minden induláskor. Ez azért fontos, mert a módosítások csak akkor lépnek életbe, ha újraindítjuk a syslog démont. Az újraindításhoz a service syslog-ng restart parancs kiadása szükséges. A konfigurációs fájl logikus felépítésű, és könnyen értelmezhető. Alapvetően három elemmel találkozhatunk benne, ezek a source, a filter és a destination. Minden üzenethez tartozik egy vagy több forrás és cél, illetve tartozhat hozzá egy vagy több szűrő. A forrás lehet például egy syslog kliens, amely hálózaton keresztül küldi az adatokat. Az adatok fogadásához hozzá kell adnunk egy forrást. A következő példa egy egyszerű hálózati forrás. Ebben az esetben a forrás neve tcp_src, a 91.83.45.77-es IP címen, és az 514-es porton fogadja az üzeneteket, a kommunikációhoz pedig TCP protokollt használ. Központosított naplózó rendszerek esetén mindenképpen szükséges egy hasonló bejegyzés a szerver oldalon.

```
source tcp_src {
        tcp(ip("91.83.45.77")
        port(514));
};
```

A lehetséges források listáját a 3.4-es ábrán látható táblázat tartalmazza.

Név	Leírás
internal()	Belső syslog-ng üzenetek.
file()	Megynyitja a megadott fájlt, és beolvassa az üzeneteket.
pipe(), fifo	Megynyitja a megadott csővezetéket, és beolvassa az üzeneteket.
program()	Megnyitja a megadott programot, és a szabványos kimenetéről olvassa az adatokat.
sun-stream(), sun-streams()	Megnyitja a megadott Streams eszközt Solaris rendszeren, és beolvassa az érkező adatokat.
syslog()	Bejövő üzenetekre vár IETF-syslog formátumban.
tcp(), tcp6()	Megadott TCP porton hallgatózik, BSD-syslog protokollnak megfelelő adatokat vár.
udp(), udp6()	Megadott UDP porton hallgatózik, BSD-syslog protokollnak megfelelő adatokat vár.
unix-dgram()	Megnyitja a megadott unix socket-et Sock_Dgram módban, és várja a beérkező üzeneteket.
unix-stream()	Megnyitja a megadott unix socket-et Sock_Stream módban, és várja a beérkező üzeneteket.

3.4 ábra Syslog-ng lehetséges források listája

(*A táblázat a Balabit Kft. által kiadott dokumentum alapján készült, amely a következő címen található:*

http://www.balabit.com/sites/default/files/documents/syslog-ng-pe-v4.0-guide-admin-en.html/sources.html)

A következő elem amire a konfigurációban szükségünk van, a szűrő. Ha a syslog-ng konfigurációs fájlban egy log bejegyzés tartalmaz filter bejegyzést, a következő történik: a rendszer fogadja a bejegyzést a megadott forrásból, majd megvizsgálja, hogy megfelel-e a megadott szűrőknek, ha megfelel, akkor a célhoz irányítja az üzenetet, ha pedig nem, akkor eldobja. A rendszer lényege, hogy az adott forrásból beérkező bejegyzéseket szétválogathatjuk fogadó oldalon, így átláthatóbb naplófájl rendszert kapunk. Lehetőség van arra is, hogy a kliens oldalon hozzunk létre szűrőket. Ez akkor hasznos, ha például nem szeretnénk minden naplóbejegyzést a szerverre továbbítani, így csökkenthető a hálózati forgalom, és nem halmozunk fel számunkra felesleges üzeneteket a szerveren. A szűrés alapja gyakorlatilag bármilyen információ lehet, amelyet az üzenet tartalmaz. A leggyakrabban az üzenet tartalma, a küldő rendszer neve, illetve az üzenet szintje szerinti szűrést alkalmazzuk. A szűrők megadásánál használhatjuk a megszokott logikai kifejezéseket, mint például az „and”, az „or”, és a „not”. A következő egyszerű példában a kliens1 gépről érkező üzeneteket szűrjük, amelyek tartalmazzák az „sshd” kifejezést. A kifejezések megadásakor reguláris kifejezéseket is használhatunk.

Filter pelda {host(„kliens1”) and match(„sshd”);};

Az utolsó elem amire szükség van egy log bejegyzésben a syslog-ng konfigurációs fájlban, a cél megadása. A cél, ahova az üzenet továbbításra kerül, ha megfelel a szűrőnek. A lehetséges célok listáját a 3.5-ös ábrán látható táblázat tartalmazza.

Név	Leírás
file()	Adott fájlba írja az adatokat.
logstore()*	Bináris logstore fájlba írja az adatokat (csak syslog-ng PE-ben érhető el).

Név	Leírás
fifo(), pipe()	Csővezetékbe írja az adatokat.
program()	Elindít egy megadott programot, majd az üzeneteket a szabványos bemenetére küldi.
sql()	SQL adatbázisba küldi az adatokat.
syslog()	Egy távoli gépre küldi az adatokat IETF-syslog protokoll használatával.
tcp() and tcp6()	Egy távoli gép megadott portjára küldi az adatokat BSD-syslog protokoll használatával TCP-n keresztül.
udp() and udp6()	Egy távoli gép megadott portjára küldi az adatokat BSD-syslog protokoll használatával UDP-n keresztül.
unix-dgram()	Egy adott unix socket-re küldi az adatokat Sock_dgram formában(BSD).
unix-stream()	Egy adott unix socket-re küldi az adatokat Sock_dgram formában(linux).
usertty()	Az adatokat egy adott terminálra küldi.

3.5 ábra Syslog-ng lehetséges célok listája

(*A táblázat a Balabit Kft. által kiadott dokumentum alapján készült, amely a következő címen található:* http://www.balabit.com/sites/default/files/documents/syslog-ng-pe-v4.0-guide-admin-en.html/destinations.html)

A következő cél bejegyzés a kliens gépeken használható, az üzenetek továbbításához a szerverre.

destination logserver { tcp("91.83.45.79" port(1468)); };

A bemutatott három elemet a konfigurációs fájlban egy log bejegyzésben kell megadnunk. Egy ilyen példa bejegyzés a következőképpen írható fel:

log{ source(forras); filter(szuro); destination(cel);};

A syslog-ng konfigurációját részletesen bemutató dokumentum a http://www.balabit.com/support/documentation/syslog-ng-pe-v4.0-guide-admin-en_0.pdf címen érhető el, pdf formátumban.

3.6. Syslog-ng teszt

A Syslog-ng tesztelését a Novell Sentinel Log Managerhez hasonlóan végeztem. A szerver és a kliensek hardver adottságai megegyeznek a Sentinel Log Manager tesztelésekor használtakkal. Mivel a Syslog-ng a Sentinel Log Managerrel szemben nem érhető el kész virtuális gépként, így egy Suse Linux Enterprise Server 11-es verzióra telepítettem a Syslog-ng Premium Edition 4.0 30 napos próbaverziót. A próbaverzió teljes funkcionalitást kínál a 30 napos időszak alatt, így a teszt szempontjából teljesen megegyező a teljes verzióval. A két kliens közül az egyiken syslog-ng fut, míg a másikon rsyslog. A teszt során ebben az esetben is TCP alapú átvitelt választottam, az eredmények titkosított adatfolyam esetén eltérőek lehetnek. Mivel a Syslog-ng a Sentinel Log Managerrel szemben nem rendelkezik grafikus felülettel, nem látható az EPS érték a szerveren, ezért a méréshez a következő parancsot használtam:
grep hh:mm:ss /var/log/messages | wc -l
A parancs meghatározza, hogy egy adott másodpercben hány üzenet került a messages naplófájlba. A parancsban a hh:mm:ss helyén a mérés idejéből kiválasztott 10 másodperces időintervallumot használtam. Tehát a parancsot tízszer futtattam, először a wc -l parancs nélkül, hogy meggyőződjek arról, hogy csak az általam generált tesztüzenetek kerültek a szűrésbe. Ezután a 10 mintavétel eredményét átlagoltam, így megkaptam a 10 másodpercre eső átlag EPS értéket a szerveren. A teszt során a második kliens gép használata előtt szinkronizálnom kellett a két kliens gép óráját, hogy az előző parancs megfelelő kimenetet adjon. A szinkronizációhoz az ntpdate parancsot használtam.
Az eredményeken jól látható, hogy a memóriahasználatot itt sem befolyásolja az eseményszám. A Syslog-ng magasabb eseményszám esetén is nagyon minimális terhelést mutat. A Balabit Kft. 2011.01.27-én kiadott egy dokumentumot, amely a Syslog-ng PE 4.0.1 általuk végzett teszteredményeit tartalmazza. A dokumentum szerint

a szerver képes volt másodpercenként 180 000 üzenet fogadására, és tárolására egyszerű szövegként. Titkosított adatfolyam használata esetén 50 000 üzenetet tudott fogadni másodpercenként, amely kevesebb, mint a harmada, a TLS nélkül mért értéknek, azonban többszöröse a Novell által ígért 7500-as másodpercenkénti eseményszámnak. Természetesen a használt szűrők, illetve az egyedi beállítások tovább csökkenthetik a teljesítményt. A Balabit Kft. által kiadott dokumentum a http://www.balabit.com/support/documentation/syslog-ng-pe-v4.0-whitepaper-performance-en.pdf címen érhető el. Az általam végzett teszt eredményeit a 3.6-os ábrán látható táblázat mutatja.

Sleep idő (s)	**Kliensek száma**	**Load a szerveren**	**EPS érték a szerveren**	**Szerver CPU terhelés (%)**	**Szabad mem.(MB)**
-	0	0	0	0,00%	13 MB
0,1	1	0	8	0,00%	24 MB
0,01	1	0	67	0,00%	24 MB
Nincs korlát	1	0,02	580	0,3%	23 MB
0,01	2	0,01	121	0,1%	20 MB
Nincs korlát	2	0,05	680	0,4%	18 MB
Nincs korlát	4	0,8	1741	0,5%	30 MB

3.6 ábra Syslog-ng teszt eredmények

3.7. Syslog-ng tárhely foglalás becslése

Naplózó rendszer tervezésekor jó, ha meg tudjuk becsülni, hogy mekkora tárhelyre lesz szükségünk. A Novell Sentinel Log Managerben erre egy beépített funkció szolgál, amely mutatja a becsült tárhely foglalást a következő 90 napra. A Syslog-ng esetén nekünk kell megbecsülni a tárhely foglalást. A szükséges helyigény kiszámításához szükségünk lesz arra, hogy hány naplóüzenetet fogad átlagosan a rendszerünk, egy

naplóüzenet átlagosan mekkora méretű, hány napig szeretnénk egyszerű szövegként, és tömörített formátumban tárolni az üzeneteket, illetve, hogy milyen arányú tömörítésre képes az általunk használt program. Az átlagos másodpercenkénti üzenetszám minden rendszer esetén más, azonban a 3.6-os fejezetben használt módszerrel könnyen kiszámítható. Fontos, hogy minél nagyobb időintervallumban, és minél több mintavétellel számoljunk, hogy az adat a legjobban közelítsen a valósághoz. Az egyes naplóbejegyzések átlagos mérete eltérő lehet különböző rendszereken, nagyban függ a használt programoktól. Összegyűjtöttem többféle naplóállományt különböző rendszerekből, az egyes naplóállományok méretét elosztottam a bennük található bejegyzések számával, majd a kapott értékeket átlagoltam, így megkaptam, hogy egy naplóbejegyzés átlagos mérete körülbelül 146 byte. A naplóállományok tömörítés szempontjából jó adottságokkal rendelkeznek. Amennyiben eldöntöttük, hogy milyen tömörítő programot fogunk alkalmazni, teszteljük a rendszeren, hogy átlagosan milyen arányban képes tömöríteni a naplóállományokat. Én bzip2-t használtam, ebben az esetben 1:15 arányú tömörítést sikerült elérnem átlagosan, ami igen jónak mondható. Ezután azt kell eldöntenünk, hogy hány napig szeretnénk egyszerű szövegként, illetve hány napig szeretnénk tömörítve tárolni a fájlokat. Amennyiben minden adat rendelkezésre áll, a 3.7-es ábrán látható képlettel megbecsülhető a tárhely foglalás:

$$(aEPS * 86400 * day_1 * aSize) + (aEPS * 86400 * day_2 * aSize) * aComp = usage(byte)$$

Ahol:

aEPS=átlagos eseményszám másodpercenként

day_1=napok száma, ameddig az adatot egyszerű szövegként szeretnénk tárolni

aSize=naplóüzenetek átlagos mérete byte-ban megadva

day_2=napok száma, ameddig az adatokat tömörítve szeretnénk tárolni

aComp=átlagos tömörítési arány(bzip esetén ez 1/15)

usage=tárhely igény byte-ban

3.7 ábra Syslog-ng tárhely számítás

4. Ingyenes kiegészítő programok

4.1. Stunnel

Amennyiben olyan naplózó rendszert használunk, amely nem támogatja a titkosított adatátvitelt, jó megoldás lehet a stunnel használata. A stunnel egy ingyenes program, amely egy biztonságos kommunikációs csatornát hoz létre két gép között TLS használatával. Az alkalmazási réteg számára transzparens, így bármilyen alkalmazás esetén használható, amely TCP protokollt használ az adatok átviteléhez. A stunnel működési elve az, hogy az alkalmazások, amelyek nem képesek titkosított adatfolyamot továbbítani, nem közvetlenül egymással kommunikálnak, hanem egy lokális stunnel portra küldik az adatot. Például, ha egy kliens és egy szerver között szeretném továbbítani a syslog által generált üzeneteket TLS segítségével, akkor a következő beállításokra lesz szükség:

- Először mindkét oldalon telepíteni kell a stunnel programot. A stunnel linux és windows rendszereken is ingyenesen elérhető.
- A szerveren rendelkezni kell egy X.509 tanusítvánnyal.
- A szerver stunnel konfigurációjában meg kell adni a tanusítvány elérési útját, illetve, hogy melyik porton hallgatózzon, és a fogadott adatokat melyik porton továbbítsa. Például a 50514-es porton érkező adatokat, továbbítsa az 514-es porton hallgatózó syslog szerver felé. Az adatok itt már titkosítatlan formában továbbítódnak, azonban ez már lokális kommunikáció.
- A kliens stunnel konfigurációjában meg kell adni, hogy a lokálisan az 514-es portra küldött adatokat a stunnel szerver 50514-es portjára küldje tovább.
- A kliens syslog konfigurációban módosítani kell a cél gép IP címét localhost-ra.

A kommunikáció így titkosított lesz, bár a kliens nem authentikált. A stunnel részletes dokumentáció, és a gyakran használt programok beállításának leírása megtalálható a

http://stunnel.org/?page=docs oldalon. A stunnel hátránya, hogy külön telepíteni és konfigurálni kell minden egyes kliensen és szerveren, ami egy gyakran változó, illetve sok eszközt tartalmazó környezet esetében jelentős plusz munkával jár.

4.2. Logwatch

A Logwatch egy ingyenes naplóelemző, jelentéskészítő szoftver. Amennyiben nincs szükségünk hivatalos jelentés készítésére, azonban szeretnénk napi áttekintő adatokat kapni a rendszerről, a Logwatch jó kiegészítő megoldást jelent. A legfontosabb funkciói a következők:

- Testreszabható jelentéskészítés
- Egyszerű szöveg, és HTML alapú jelentés
- E-mail küldés
- Jelentés mentése fájlba

További információ a Logwatch hivatalos oldalán található a http://www.logwatch.org címen.

4.3. Logrotate

A Logrotate egy naplófájl rendszerező, tömörítő eszköz. Segítségével megadhatjuk, hogy bizonyos naplófájlokat, adott időközönként nevezzen át például dátum szerint, tömörítse, vagy küldje el e-mailben. Beállítható, hogy ne csak bizonyos időközönként végezze el a szükséges műveleteket, hanem akkor is, ha a fájl mérete meghaladja a megadott határt. A Logrotate man oldala a http://linuxcommand.org/man_pages/logrotate8.html címen található, az egyes kapcsolók és a konfiguráció leírása itt olvasható.

5. Összegzés

A naplózórendszer kiválasztásánál először is azt kell eldöntenünk, hogy mire szeretnénk használni. Fel kell sorolni azokat a funkciókat amelyekre szükségünk lehet, és az

alapján dönteni, hogy melyik szoftver elégíti ki jobban ezeket az igényeket. A dolgozatom eddigi részében bemutattam a két rendszer legfőbb funkcióit, illetve felépítettem egy teszt környezetet.

Egy rendszer kiválasztásánál fontos szempont lehet a termék ára. A Syslog-ng ingyenes változatával, a korábban bemutatott kiegészítő programokkal, illetve a shell script írás alapjainak birtokában egy kis- vagy középméretű rendszer általános naplózási feladatai tökéletesen elláthatók. Fontos figyelembe venni azt is, hogy az ingyenes verziókhoz nem jár terméktámogatás, azonban mindkét bemutatott termékhez széleskörű dokumentáció érhető el a gyártó oldalán.

A Novell Sentinel Log Manager egy egyszerűen használható, teljes körű naplómenedzsment eszköz, amely alkalmas heterogén rendszerek naplózására. A rendszer kiépítéséhez, telepítéséhez alapszintű ismeretek szükségesek, a telepítés után az általános feladatok elvégezhetők a grafikus felületen. Nem csak gyűjti a naplófájlokat, hanem adatbázisban tárolja, archiválja őket, és jelentéseket készít belőlük. A hátrányai közé tartozik a magasabb hardver követelmény, és a korlátozott eseményszám, amely maximum 7500 lehet. Az ingyenes verzió ugyan teljes funkcionalitást kínál, azonban a másodpercenkénti 25 eseményt egy középméretű rendszer is könnyen átlépi.

A Syslog-ng esetében az egyetlen változat, amely grafikus felületet, jelentéskészítést, és egyéb funkciókat kínál, a célhardverrel együtt megvásárolható StoreBox változat. A csak szoftver alapú megoldások nem kínálnak teljes menedzsment megoldást. Az archiválás, keresés, jelentés készítés feladatait egyéb módon kell megoldanunk, illetve nem áll rendelkezésre grafikus felület. A kisebb funkcionalitást a hatalmas teljesítmény ellensúlyozza. A Syslog-ng esetében nincs korlátozva az eseményszám, így egy szokatlanul nagy számú bejegyzést generáló rendszer esetében jó megoldást jelenthet.

A két rendszer számos dologban eltér egymástól, ezért az alapján kell választanunk, hogy az adott célrendszer feladatainak ellátására melyik alkalmasabb. A Novell Sentinel Log Manager fontosabb előnyeit és hátrányait felsoroló táblázat az 5.1-es ábrán látható, a Syslog-ng előnyeit és hátrányait leíró táblázat pedig az 5.2-es ábrán található.

Novell Sentinel Log Manager tulajdonságai

Előnyök	Hátrányok
Egyszerű telepítés	A licensz eseményszámhoz kötött
Egyszerű konfiguráció	Ingyenes verzió csak 25 EPS
Grafikus felületű menedzsment	Csak 64 bit-es architektúrát támogat
Átlátható hierarchikus kép a rendszerről	Magasabb hardverkövetelmény
Helyfoglalási statisztika, előrejelzés	Syslog-ng-vel szemben kisebb eseményszám
Elosztott keresés	
Automatikus archiválás	
Keresés archivált adatokban	
Jelentés készítés előírásoknak megfelelően	
Tárolás adabázisban, akár 10:1 arányú tömörítéssel	
Heterogén rendszerek támogatása	

5.1 ábra Novell Sentinel Log Manager fontosabb előnyei és hátrányai

Syslog-ng PE 4.0 tulajdonságai	
Előnyök	**Hátrányok**
Egyszerű telepítés	Nincs tömörítés
Bonyolultabb konfiguráció, de jobban	Archiválás nincs alapból, logrotate, vagy

testreszabható	script segítségével lehet
Nagyobb eseményszám	Kereséshez alapvető linux ismeret szükséges
32 bit-es architektúrát is támogat	Grafikus felület csak StoreBox esetén
Plaintext és adatbázis tárolás	Jelentés készítés csak StoreBox-al
Ingyenes változat is jól használható	

5.2 ábra Syslog-ng fontosabb előnyei és hátrányai

6. Irodalomjegyzék

[1] Marcel Gagné:Linux Rendszerfelügyelet, Kiskapu kiadó, 2002, 612. oldal

[2] Request for Comments: 5246 The Transport Layer Security (TLS) Protocol

[3] Request for Comments: 4158 Internet X.509 Public Key Infrastructure

[4] Request for Comments: 3164 The BSD syslog Protocol

[5] Request for Comments: 3195 Reliable Delivery for syslog

[6] Request for Comments: 5424 The Syslog Protocol

7. Mellékletek

223/2009 X.14 kormány rendelet 15. paragrafusa:

- **15. §** (1) A szolgáltatást nyújtó szervezet az általa működtetett rendszerben vagy annak környezetében vagy mindkettőben gondoskodik a rendszer működése szempontjából meghatározó folyamatok valamennyi kritikus eseményének naplózásáról.
- (2) A szolgáltatást nyújtó szervezet a naplózandó események körét, a napló adattartalmának megőrzési idejét - a vonatkozó jogi szabályozás alapján, az adott eljárási cselekmény biztonsági jellegére, érzékenységére tekintettel - határozza meg. A megőrzési időn belül a megbízhatóság megítéléséhez szükséges mértékben valamennyi, az eljárási cselekménnyel kapcsolatos eseménynek rekonstruálhatónak kell lennie. Naplózni kell minden személyes adat továbbítását.
- (3) A naplóállomány bejegyzéseit védeni kell az arra jogosulatlan személy általi hozzáféréstől, módosítástól, törléstől, illetve biztosítani kell, hogy a napló tartalma a megőrzési időn belül a jogosult számára megismerhető és értelmezhető maradjon.
- (4) A naplóállományokat a 16-17. §-ban szabályozott mentési rendnek megfelelően, a maradandó értékű dokumentumokra vonatkozó szabályok szerint kell tárolni, hogy egy esetleges lokális károsodás ne tegye lehetetlenné a bizonyítást.
- (5) A naplóállományok megőrzési idejét - a (2) bekezdésben foglaltak figyelembevételével - a vonatkozó iratkezelési szabályzatok részeként kell meghatározni. A működtető a vonatkozó jogszabály, illetve iratkezelési szabályzat rendelkezésétől függően, a megőrzési határidő lejártával gondoskodik a naplóállományok adathordozóinak levéltári őrizetbe adásáról vagy az adatállományok dokumentált, visszaállítást kizáró megsemmisítéséről.

1996-os CXII. törvény 13/C paragrafusa

13/C. § (1) A pénzügyi intézménynek ki kell alakítania a pénzügyi, kiegészítő pénzügyi szolgáltatási tevékenységének ellátásához használt informatikai rendszer

biztonságával kapcsolatos szabályozási rendszerét és gondoskodnia kell az informatikai rendszer kockázatokkal arányos védelméről. A szabályozási rendszerben ki kell térni az információtechnológiával szemben támasztott követelményekre, a használatából adódó biztonsági kockázatok felmérésére és kezelésére a tervezés, a beszerzés, az üzemeltetés és az ellenőrzés területén.

- (2) A pénzügyi intézmény köteles az informatikai rendszer biztonsági kockázatelemzését szükség szerint, de legalább kétévente felülvizsgálni és aktualizálni.
- (3) Az informatika alkalmazásából fakadó biztonsági kockázatok figyelembevételével meg kell határozni a szervezeti és működési rendeket, a felelősségi, nyilvántartási és tájékoztatási szabályokat, a folyamatba épített ellenőrzési követelményeket és szabályokat.
- (4) A pénzügyi intézménynek ki kell dolgoznia az informatikai rendszerének biztonságos működtetését felügyelő informatikai ellenőrző rendszert és azt folyamatosan működtetnie kell.
- (5) A biztonsági kockázatelemzés eredményének értékelése alapján a biztonsági kockázattal arányos módon gondoskodni kell legalább az alábbiakról:
- *a)* a rendszer legfontosabb elemeinek (eszközök, folyamatok, személyek) egyértelmű és visszakereshető azonosításáról,
- *b)* az informatikai biztonsági rendszer önvédelmét, kritikus elemei védelmének zártságát és teljeskörűségét biztosító ellenőrzésekről, eljárásokról,
- *c)* a rendszer szabályozott, ellenőrizhető és rendszeresen ellenőrzött felhasználói adminisztrációjáról (hozzáférési szintek, egyedi jogosultságok, engedélyezésük, felelősségi körök, hozzáférés naplózás, rendkívüli események),
- *d)* olyan biztonsági környezetről, amely az informatikai rendszer működése szempontjából kritikus folyamatok eseményeit naplózza és alkalmas e naplózás rendszeres (esetleg önműködő) és érdemi értékelésére, illetve lehetőséget nyújt a nem rendszeres események kezelésére,
- *e)* a távadatátvitel bizalmasságáról, sértetlenségéről és hitelességéről,
- *f)* az adathordozók szabályozott és biztonságos kezeléséről,
- *g)* a rendszer biztonsági kockázattal arányos vírusvédelméről.

- (6) A pénzügyi intézménynek tevékenysége ellátásához, nyilvántartásai naprakész és biztonságos vezetéséhez meg kell valósítania a biztonsági kockázatelemzés alapján indokolt védelmi intézkedéseket és rendelkeznie kell legalább a következőkkel:
- *a)* informatikai rendszerének működtetésére vonatkozó utasításokkal és előírásokkal, valamint a fejlesztésre vonatkozó tervekkel,
- *b)* minden olyan dokumentációval, amely az üzleti tevékenységet közvetlenül vagy közvetve támogató informatikai rendszerek folyamatos és biztonságos működését - még a szállító, illetőleg a rendszerfejlesztő tevékenységének megszűnése után is - biztosítja,
- *c)* a szolgáltatások ellátásához szükséges informatikai rendszerrel, valamint a szolgáltatások folytonosságát biztosító tartalék berendezésekkel, illetve e berendezések hiányában az ezeket helyettesítő egyéb - a tevékenységek, illetve szolgáltatások folytonosságát biztosító - megoldásokkal,
- *d)* olyan informatikai rendszerrel, amely lehetővé teszi az alkalmazási környezet biztonságos elkülönítését a fejlesztési és tesztelési környezettől, valamint a megfelelő változáskövetés és változáskezelés fenntartását,
- *e)* az informatikai rendszer szoftver elemeiről (alkalmazások, adatok, operációs rendszer és környezetük) olyan biztonsági mentésekkel és mentési renddel (mentések típusa, módja, visszatöltési és helyreállítási tesztek, eljárási rend), amelyek az adott rendszer helyreállíthatóságát a rendszer által nyújtott szolgáltatás kritikus helyreállítási idején belül lehetővé teszik. Ezen mentéseket kockázati szempontból elkülönítetten és tűzbiztos módon kell tárolni, valamint gondoskodni kell a mentések forrásrendszerrel azonos szintű hozzáférés védelméről,
- *f)* jogszabályban meghatározott nyilvántartás ismételt előhívására alkalmas adattároló rendszerrel, amely biztosítja, hogy az archivált anyagokat a jogszabályokban meghatározott ideig, de legalább öt évig, bármikor visszakereshetően, helyreállíthatóan megőrizzék,
- *g)* a szolgáltatásai folyamatosságát akadályozó rendkívüli események kezelésére szolgáló tervvel.
- (7) A pénzügyi intézménynél mindenkor rendelkezésre kell állnia:

- *a)* az általa fejlesztett, megrendelésére készített informatikai rendszer felépítésének és működtetésének az ellenőrzéséhez szükséges rendszerleírásoknak és modelleknek,
- *b)* az általa fejlesztett, megrendelésére készített informatikai rendszernél az adatok szintaktikai szabályainak, az adatok tárolási szerkezetének,
- *c)* az informatikai rendszer elemeinek a pénzügyi intézmény által meghatározott biztonsági osztályokba sorolási rendszerének,
- *d)* az adatokhoz történő hozzáférési rend meghatározásának,
- *e)* az adatgazda és a rendszergazda kijelölését tartalmazó okiratnak,
- *f)* az alkalmazott szoftver eszközök jogtisztaságát bizonyító szerződéseknek,
- *g)* az informatikai rendszert alkotó ügyviteli, üzleti szoftvereszközök teljes körű és naprakész nyilvántartásának.
- (8) A szoftvereknek együttesen alkalmasnak kell lenni legalább:
- *a)* a működéshez szükséges és jogszabályban előírt adatok nyilvántartására,
- *b)* a pénz és az értékpapírok biztonságos nyilvántartására,
- *c)* a pénzügyi intézmény tevékenységével összefüggő országos informatikai rendszerekhez történő közvetlen vagy közvetett csatlakozásra, ideértve a pénzforgalmi számlák cégbíróság felé történő bejelentését is,
- *d)* a tárolt adatok ellenőrzéséhez való felhasználására,
- *e)* a biztonsági kockázattal arányos logikai védelemre és a sérthetetlenség védelmére.
- (9) A pénzügyi intézménynek belső szabályzatában meg kell határoznia az egyes munkakörök betöltéséhez szükséges informatikai ismeretet.
- (10) Az (1)-(9) bekezdésben foglaltaknak a pénzügyi, kiegészítő pénzügyi szolgáltatási tevékenységéhez kapcsolódóan a pénzforgalmi intézménynek és az elektronikuspénz-kibocsátó intézménynek is meg kell felelnie.

Syslog-ng három változatának összehasonlítása

	syslog-ng OSE	**syslog-ng PE**	**syslog-ng Store Box**
Üzenetek megbízható továbbítása (TCP)	✔	✔	✔
Üzenetek szűrése tartalom alapján	✔	✔	✔
Adatbázis táblák, könyvtárak, fájlok dinamikus létrehozása makrók segítségével	✔	✔	✔
IPv6 támogatás	✔	✔	-
Naplózás közvetlenül adatbázisba	✔	✔	✔
Üzenetek titkosított továbbítása (TLS)	✔	✔	✔
A legfrissebb IETF-syslog szabványok támogatása	✔	✔	✔
Üzenetek feldolgozása és módosítása	✔	✔	✔
Titkosított, aláírt, időpecséttel ellátott tárolás	-	✔	✔
Üzenetek pufferelése merevlemezre	-	✔	✔
Többsoros üzenetek kezelése	-	✔	-
Magas rendelkezésre állás kliensoldali támogatása	-	✔	-
Üzenetek címkézése	✔	✔	✔
Üzenetek azonosítása és osztályozása mintaadatbázis alapján	✔	✔	✔
Név-érték párok kinyerése az azonosított üzenetekből	✔	✔	✔
Tetszőleges metaadat hozzáfűzése az azonosított üzenetekhez	✔	✔	✔

	syslog-ng OSE	**syslog-ng PE**	**syslog-ng Store Box**
Üzenetek korrelálása valós időben	✔	-	-
Azonosított üzenetekhez események kiváltása	✔	-	-
Linux process accounting üzenetek gyűjtése	✔	-	-
Üzenetküldés sebességének szabályozása	✔	✔	✔
Részletes statisztikák gyűjtése a feldolgozott üzenetekről például forrás, cél, stb. alapján	✔	✔	✔
Windows támogatás	-	✔	✔
Hardver berendezés	-	-	✔
Web alapú konfigurációs felület	-	-	✔
Magas rendelkezésre állás támogatása	-	-	✔
Integrált naplókereső és böngésző felület	-	-	✔
Testreszabható riportok készítése	-	-	✔

A táblázat forrása: http://www.balabit.com/hu/network-security/syslog-ng/opensource-logging-system/features/comparison