КЫРГЫЗ-ТҮРК МАНАС УНИВЕРСИТЕТИ
ТАБИГЫЙ ИЛИМДЕР ИНСТИТУТУ
КОМПЬЮТЕР ИНЖЕНЕРИЯ БАГЫТЫ

SQUID ПРОКСИ СЕРВЕР НЕГИЗИНДЕ ВЕБ ТРАФИКТИН ОНЛАЙН МОНИТОРИНГИ ЖАНА ЭСЕП СИСТЕМИ.

(Squid proxy server bazında web trafiginin online gözlenmesi ve hesab sistemi.)

(МАГИСТРДИК ДИССЕРТАЦИЯ)

Тилек МАЙТЫКОВ

БИШКЕК 2010

КЫРГЫЗ-ТҮРК МАНАС УНИВЕРСИТЕТИ
ТАБИГЫЙ ИЛИМДЕР ИНСТИТУТУ
КОМПЬЮТЕР ИНЖЕНЕРИЯ БАГЫТЫ

Squid прокси сервер негизинде веб трафиктин онлайн мониторинги жана эсеп системи.

(Squid proxy server bazında web trafiginin online gözlenmesi ve hesab sistemi.)

(МАГИСТРДИК ДИССЕРТАЦИЯ)

Майтыков Тилек

Жетекчи:
док. Бакыт Шаршембаев

БИШКЕК 2010

ЧЕЧИМ

Кыргыз-Түрк Манас Университетинин Табигый Илимдер Институтунун экзамендик инструкциясынын ….-жобосунун ылайык …….№ жыйында уюшулган комиссия, компьютер инженерия бөлүмүнүн магистранты ***Squid прокси сервер негизинде веб трафиктин онлайн мониторинги жана эсеп системи.***

темасында жазган дипломдук проектин анализдеп, 2010 ж. саат ….да жактоого кабыл алынды.

Магистрант ….минута убакыт ичинде дипломдук проектин жактап, комиссия көпчүлүк добуш менен/бир добуштан Кабыл алынбайт/Кабыл алынсын/Кайра оңдолсун деген чечим чыгарылды.

Жюри Мүчөсү (жетекчи) док. Бакыт Шаршембаев Кыргыз-Түрк Манас университети	Жюри Мүчөсү _____ Кыргыз-Түрк Манас университети
Жюри Мүчөсү _____ Кыргыз-Түрк Манас университети	Жюри Мүчөсү _____ Кыргыз-Түрк Манас университети
Жюри Мүчөсү _____ Кыргыз-Түрк Манас университети	Жюри Мүчөсү _____ Кыргыз-Түрк Манас университети

...../...../ 2010

КЫСКАЧА МАЗМУУНУ

Даярдаган:	Майтыков Тилек
Университет:	Кыргыз Түрк Манас университети
Багыты:	Компьютер инженерия адистик багыты
Иштин сыпаты:	Магистрдик иш
Беттердин саны:	VII+84
Бүтүрүү датасы:	18/06/2010
Илимий жетекчи:	док. Бакыт Шаршембаев

Squid прокси сервер негизинде веб трафиктин онлайн мониторинги жана эсеп системи.

Интернеттин өнүгүүсү менен интернет-трафигинин көлөмү да чоңоюсу белгилүү. Акыркы жылдары Интернеттин экспоненциалдык өсүп өнүгүүсү байкалууда жана информациондук революциянын башаты болуп келе жатат. Күндөн күнгө көптөгөн интернет сервистер пайда болууда. Бул сервистер ар түрдүү кызмат, маалымат, тейлөөлөрдү камтышат. Ар бир сервис ар түрдөгү, ар көлөмдөгү трафикти жаратат жана бул трафиктин изилдөөсүнүн негизинде дагы сапатту, дагы илгери тейлөөлөрдү жаратуу мүмкүн. Бүгүнкү күндө интернет трафиктин чоң ролду ойноору шексиз жана ар түрдүү тараптан изилденүүдө. Мындай изилдөөлөрдүн айрым учурунда ар бир мекемедеги системдик администраторлордун жүргүзөрү белгилүү. Мекемедеги колдонуучулар арасында интернетти бирдей бөлүштүрүү, кээ бир колдонуучулардын кыянаттык жоруктарына тыюу салуу максатында интернет трафиктин изилденүүсү бул айрым учурда актуалдуу экендиги сөзсүз. Учурдагы изилдөөдө так ушул проблемаларды чечүүдө жаралган кыйынчылыктарды, инструменттерди, методторду дагы терең кароо максаты коюлган. Конкреттүү SQUID прокси сервер базасында веб трафиктин онлайн мониторинги жана эсеп системи каралган.

Ачкыч сөздөр: прокси сервер, сервер, веб трафик, анализ, мониторинг, SQUID.

ÖZ

Squid proxy server bazında web trafiginin online gözlenmesi ve hesab sistemi

İnternet geliştikçe internet trafiğinin da artması söz konusudur. Her gün yüzlerce internet servisler ortaya çıkmakta. Bu servisler kullanıcılara her türlü bilgi, servis, ürün sunuyorlar. Bu internet servisler her türlü trafik yaratmakta ve bu trafiğinin araştırılması servislerin daha kaliteli düzeye çıkarılmasına yardım eder. Bugunkü günlerde internet trafik büyük kontrol altına alınıyor ve araştırılıyor. Böyle bir internet trafiğinin gözlenmesini her büyük şirketlerin administratörlörü yapmakta. Her kullanıcılara aynı internet genişliğini sağlamak için bazı taciz kullanıcıları tespit edip sınırlamak için. İşte bu tez ortaya cıkan problemi her taraftan araştırmak ve internet trafik araçların eksiklerin göz altına alarak daga geniş özelliklere sahip olan araç geliştirmektir. Konuyu daha soyutluşturarak SQUID proxy server bazında web trafiginin online gözlenmesi ve hesab sistemi geliştiriliyor.

Anahtar Sözcükler: proxy server, access.log, web-analizatör, SQUID, Internet, ağ kullanım, monitorleme.

ABSTRACT

Squid proxy server based online web traffic observation and account system

Internet traffic grows on the internet. Every day hundreds of Internet services emerging. These services are present to users some information, services, products. All this Internet services are creating traffic and anaylizing that traffic helps to level up quality of services. Today internet traffic is taken under control and investigation. Every major companies administrator is monitoring Internet traffic to ensure that all users share web chanel width and to detect and limit some abusive users. This thesis emerged from all sides of internet monitoring problem and trying to develop tool taking into account the missing features of exists solutions. To be more precisly we are going to use SQUID proxy server as a base and create online account system and monitoring of internet traffic tool.

Keywords: proxy server, access.log, web-analizator, SQUID, Internet, network utilization, monitoring.

КЫСКАРТУУЛАР

SQUID	Прокси сервердин аты
SARG	Лог-анализатордун аты
SAMS	Лог-анализатордун аты
FreeSA	Лог-анализатордун аты
MySAR	Лог-анализатордун аты
SQSTAT	Cache manager протоколун колдонгон анализатор
LightSquid	Лог-анализатордун аты
NFS	Network file system
DNS	Domain name system
IP/TCP	Internet protocol / Transmission Control Protocol
LDAP	Lightweight Directory Access Protocol
PAM	Protocol Analysis Module
SMB	Samba
NTLM	NT LAN Manager
FTP	File transfer protocol
HTTP	Hyper text transfer protocol
SOCKS	Socket protocol
NAT	Network Address Translation
UML	Unified Modeling language

ТАБЛИЦА, ГРАФИК, СҮРӨТТӨР

Маазмуну

Кириш

Биринчи бөлүмдө прокси сервер түшүнүгү, аткарган функциялары жана учурдагы күндө жаралган проблемалар каралган. Прокси серверлердин түрлөрү, тейлөө маселери, кемчиликтери жана артыкчылыктары боюнча салыштырылган. Айрыкча SQUID прокси сервери үстүндө токолуп, ички структурасы каралат.

Ал эми экинчи бөлүмдө учурда бар болгон анализаторлор үстүндө изилдөө жүргүзүлөт, жана тесттик компьютерде салыштырмалуу сыноодон өткөрүлүшөт. Бул сыноонун жыйынтыктары график жана таблица түрүндө келтирилет. Жыйынтыктын үстүндө анализ жүргүзүп дипломдук иштин максатына жараша жеңип чыккан анализаторлор белгиленип кийинки кадамдарда активдүү колдонулушат.

Жогорку бөлүмдөрдүн чыгарылаган жыйынтыктардын негизинде жана SQUID прокси сервердин мүмкүнчүлүктөрүн колдонулуп бар болгон анализаторлордун базасында прототиптер иштетилип чыгат. Бул прототиптердин өзгөчөлүктөрү жана кемчиликтерин карайбыз. Айрыкча SQUID прокси сервердин “cache_manager” протоколунда токтолобуз. Бул протокол адбан бай маалыматка ээ болгону менен учурда “cache_manager”протоколу менен иштеген абдан аз инструменттер бар болгондугун көрө алабыз. Бул протокол көбүнчө SQUID прокси серверди жараткан программистер тарабынан колдонулулат. Жана протокол өндүрүш прокси серверлерде тилекке каршы унутулуп калынат же болбосо такыр өчүрүлөт. Cache manager протоколуна өзгөчө көнүл коюлат жана учурдагы изилдөөнүн проблемасына жаракту чечим болгондугу белгиленет.

Бүгүнкү күндө веб трафиктин көптөгөн анализаторлору бар. Кээбир трафик анализаторлор өз алдынча система түрүндө иштешет, мисал катары UserGate[1], LabrisWebfilter[2] программалары (ичинде прокси сервер функциясы бар). Мындан сырткары веб трафикти анализдөө үчүн прокси сервердин лог

[1] www.entensys.com

[2] http://www.labris.eu

файлдарын анализдөө принцибине негизделген популярдуу программалар бар, мисал катары SQUID[3] + SARG[4] программалары.

Өз алдынча иштеген трафик анализаторлор (UserGate, LabrisWebfilter) кошумча тейлөө талап кылышат, ички код жабык болот, жана жаңы түрдөгү отчет жаратуу үчүн фирмага кайрылуу зарыл. Бирок коду ачык прокси серверлер бар, жана алардын жараткан протоколдук файлдары да ачык, жана бул файлдын негизинде оңой эле отчет түзүү мүмкүн. Мисал катары SQUID прокси серверде бир нече лог-анализаторлор жазылган. Көбүнчөлөрү сервер администраторлор тарабындан жазылып колдон колго өтүп жайылып таралган.

Проблема

Бүгүнкү күндө бар болгон анализаторлордун үстүндө изилдөө жүргүзүп төмөнкү проблемага туш келдик.

- Веб трафик отчетунун бир нече убакыт кечигүүсү (лог файлдын көлөмүнүн чоңдугунан көбүнчө лог-анализаторлор бир нече минутадан – бир нече саатка чейин иштеши мүмкүн)
- Веб-интерфейс жоктугу (Web-интерфейс аркылуу прокси сервердин жана анализатордун параметрлерин өзгөртүүгө мүмкүнчүлүк жок болгондуктан, түздөн түз конфигурациялык файлды билип, таап, өзгөртүү зарыл)
- Отчеттун сактоо жана башкаруу түрү (көб учурда жөнөкөй html-файл түрүндө сакталат, ошондуктан сактоодо чон ыңгайсыздыктарга алып келет.)
- Заматта (онлайн) мониторинг жүргүзүү мүмкүнчүлүгүнүн жоктугу

Прокси сервердин лог файлы орточо 500 мегабайт - 2 гигабайт чейин өсүшү мүмкүн [1]. Кээбир чоң мекемелерде бир суткада 8-10 гигабайтка чейин лог файлдын өсүшү байкалат Бул лог файлды иштетүү үчүн анализаторлор бир нече саат керектирет. Администраторлор анализаторду түнкү убакытка иштетип коюшат жана эртеси күнү гана маалыматты ала алышат жана заматта

[3] www.squid-cache.org

[4] http://sarg.sourceforge.net

(онлайн) трафик жөнүндө маалымат алуу кыйынчылыгына алып келет. Бул проблеманы cron-утилитасы жардамы менен анализаторду ар бир минута сайын чакыруу менен чечүүгө аракет жазашат. Учурдагы иштин максаты бар болгон SQUID прокси серверинин анализаторлорун изилдөө жана жаралган кыйнчылыктардын эффективдүү чечүү жолун табуу.

БӨЛҮМ-1 ПРОКСИ СЕРВЕР ЖӨНҮНДӨ КЫСКАЧА ТҮШҮНҮК

SQUID прокси сервердин пайда болушунун себеби Интернеттин популярдулугу жана бул себебтен колдонуучулардын күндөн күнгө көбөйүшү, сырткы IP (external IP) адрестердин баардык колдонуучуларга жетишсиздиги, коопсуздук суроосунун өсүшү болгон. IP4 протоколу боюнча саналуу[5] сандагы гана IP адрес болушу мүмкүн (бул жерде 192.168.0.0/16, 172.16.0.0/12 жана 10.0.0.0/8 адрестер ички тармакта гана колдонулат). Ошондой эле сырткы IP адрестер акчалай сатылгандыгы үчүн чоң мекемелерге (бир топ компютерлери бар болгон) абдан ыңгайсыз болгон. Мына ушул проблемаларды чечүү үчүн прокси серверлер иштетилип чыгышкан. Биринчи прокси серверлердин функциялары эң эле жөнөкөй болгон. Алар болгону ички тармактагы (internal network) колдонуучуларга Интернетке байланышты сунуштаган. Прокси сөзү (proxy) англисче тилде **«ишеништүү жак», «инанымдуу жак»** деп которулат. Башкача айтканда прокси сервер колдонуучу атындан Интернетте операцияларды аткарат, жана башка Интернеттеги сервистерге өз атынан баш урат. Мындайча колдонуучулар сырткы тармактарга белгисиз болуп калат. Так айтканда колдонуучулардын IP адрестери сыртка белгисиз болот, жалгыз гана прокси серведин IP адреси сыртка көрүнөт. Прокси сервер мындайча ички колдонуучуларды жашыруу функциясын да аткарып калат, жана сырткы салгылардан куткарат (анткени ички колдуучулар жашырылган).

[5] A-class 0.0.0.0 - 127.255.255.255, B-class 128.0.0.0 – 191.255.255.255, C-class 224.0.0.0 – 239.255.255.255

Бүгүнкү күндө кеңири колдонулаган маршрутизаторлор да окшош функцияларды аткарат, жана прокси серверледин бир түрү – NAT прокси сервер болуп саналышат. Булар дагы ички колдонуучуларга Интернет сунуштайт, ошондой эле колдонуучуларды жашырат. Бирок NAT маршрутизаторлордун жөнөкөй прокси серверлеге караганда бир кемчилиги бар. Маршрутизаторлор трафикти анализдей албайт. Алар жөн гана ички колдонуучулардан келген пакеттердин IP адресин өзүнүн IP адреси менен алмаштыруу жүргүзөт, жана тескеринче сырттан келген пакеттерди өзүнүн таблицасына жараша керектүү колдонуучуна жеткирет. Дагы бир өзгөчүлүк маршрутизатор ички колдонуучуларга физикалык байланышууну талап кылат. Башкача айтканда байланыш кабелдер түздөн түз маршрутизаторго байланыш керек. Ал эми прокси сервер болсо Интернеттин каалаган чекитинде орнотулушу мүмкүн, жана ички куралдарга туура келген колдонуучуга Интернет байланышын сунуштайт. Прокси серверлер маршрутизаторлорго караганда кэширлөө (cache) жүргүзө алат.

1.1 Прокси серверлердин типтери

Прокси серверлердин бир нече типтери бар, алар өз алдынча иштөө протоколдору менен айырмаланышат. Эң популярдуу прокси серверлердин типтери булар http-, Socks-, NAT-прокси серверлер. NAT прокси серверлер акыркы күндөрдө көбүнчө операциондук системалар менен курулуп келип жатышат. (Linux үй бүлөсүндө.)

HTTP прокси сервер – эң популярдуу болуп саналат. Атынан белгилүү болгондой бул прокси сервер HTTP протоколу менен иштейт, башкача айтканда браузерлер менен иштейт. Браузер проксиден керектүү барактарды талап кылганда, прокси өз атынан Интернеттен же болбосо кэштен керектүү баракты/ресурсту алып жеткирет.

Биздин изилдөөбуз HTTP протоколу менен иштеген SQUID прокси сервери болгондуктан бул типтеги проксилердин өзгөчөлүктөрүн карап кетсек:

- Кэширлөө мүмкүнчүлүгү бар. Коп колдонгон жана популярдуу баракгарды кэште сактайт мындайча сырткы трафиктин азайтуусуна жардам берет, ошондой эле баракгардын бат жүктөлүүсүнө себеп болот. Бирок бүгүн Интернет динамикалдуу[6] болгондугу үчүн көбүнчө веб-сайттар кэширлөөгө тыйю салып жатууда. Ошондуктан азыркы күндө кэширлөө менен трафиктин экономиясы максимум 15% түзөт. Эгерде чындап эле трафикти экономдоо суроосу коюлса HTTP прокси серверледи атайын бул талапка ыңгайлатып META баштыктарды каратпай койсо болот. (no-cache ачкычтык сөзүн)
- Кээбир сайтарды жабуу/тыюу мүмкүнчүлүгү бар, колдонуучулар боюнча тыюу салуу, аутентификация боюнча Интернет сунуштоо мүмкүнчүлүгү бар.
- Кээбир түрдөгү файлдардын жүктөө ылдамдыгын азайтуу жана мындайча Интернетти баардык колдонуучулар арасында бирдей бөлүп берүү мүмкүн.
- Рекламдык баннерлерди жабуу.
- Бир нече Интернет каналдар бар болсоо алардын ортосунда оптималдуу кылып трафикти бөлүштүрүү мүмкүнчүлүгү.
- Жана эң маниилүү функциялардын бири бул колдонуучулардын терең статистикасын иштетип чыгаруу.

HTTPS прокси сервер – булар жөнөкөй HTTP прокси серверлерге окшош, жөн гана кошумча шифрлөө функциясы бар. Бул типтеги прокси серверлер коопсуздукту талап кылган мекемелер колдонушат (банктар, фирмалар). Жана бул типтеги прокси серверлер өтүп жаткан трафикти анализдөө жүргүзбөйт.

FTP прокси сервер – бул типтеги проксилер FTP протоколу менен иштейт. Азыр көбүнчө HTTP проксилер FTP менен иштей алышат, бирок алар толугу менен FTP протоколундагы баардык командаларды аткара алышпайт. Көбүнчө колдонуучуларга жетиштүү функцияларды гана камтыйт. Ошондуктан эгерде

[6] PHP/ASP/C# тилдери жардамы менен динамикалдуу HTML маалымат жаратуу.

толугу менен FTP протоколу менен прокси керек болсо атайын FTP прокси серверлер колдонулат.

SOCKS прокси сервер – SOCKetS сөзүнөн келип чыккан. Атайын иштетилип чыккан SOCKS протоколу менен иштейт. Бүгүнкү күндө булар эң универсалдуу прокси серверлер болуп саналат, жана каалаган протокол менен иштөөгө мүмкүнчүлүк берет. Ошондой эле колдонуучуларды сервер тараптан аутентификация жүргүзө алат.

Булардан башка да өтө специалдуу прокси серверлер да бар, алар саналуу программалар менен иштөөгө жөндөмдүрүлгөн, жана көп учурда атайын мекемелерде гана колдонулат. Дагы бир белгилеп кетүүчү нерсе Интернетте көптөгөн прокси серверлер бар, алар каалаган колдонуучуларга прокси сервисин сунуштайт. Бул прокси серверлер «ачык прокси серверлер» же «анонимдүү прокси серверлер» деп аталышат. Каалаган колдонуучу өзүнүн жашыруусу келсе бул прокси сервер аркылуу Интернетке чыга алат.

1.2 Прокси сервер SQUID

Прокси сервер SQUID өз убагында акчалай сатылган HARVEST проектинен бөлүнүп жана энтузиасттар тарабынан өнүктүрүлүп келе жатат. SQUIDтин официалдуу сайты бул www.squid-cache.org. SQUID өтө өндүрүштүү кэширлөөчү прокси. Жана жөнөкөй колдонуучуларга абдан ыңгайлаштырылган. FTP/HTTP/HTTPS протоколдору менен иштей алат. SQUID кэширлөөчу прокси сервер болгондуктан кэширлөөчү прокси серверледин иерархиясын ICP/UDP (Internet Cache Protocol) протоколу менен түзүү мүмкүнчүлүгүн сунуштайт. Кэш катуу дискте сакталат, бирок кэш оперативдик эсте да сакталат, булар көбүнчө популярдуу DNS суроо талаптар, популярдуу интернет баракчалар.

SQUID көптөгөн аутентификация мүмкүнчүлүгүн да сунуштайт: NCSA, LDAP, MSNT, NTLM, PAM, SMB. Бул жерде MSNT жана NTLM түрлөрү Windows колдонуучуларды авторизация үчүн колдонулат.

SQUID прокси сервери замандаш Unix үй бүлөсүндөгү операциондук системалар үчүн өндүрүлгөн, бүгүнкү күндө төмөнкү операциондук системаларда иштейт:

- Linux
- FreeBSD
- NetBSD
- BSDI
- OSF and Digital Unix
- IRIX
- SunOS/Solaris
- NeXTStep
- SCO Unix
- AIX
- HP-UX
- OS/2

1.3 Прокси сервердин log- файлдары

SQUID прокси серверде лог файлдар абдан маанилүү функцияларды аткарат. Була файлдарда Squid тин иштөө каталары, системдик каталар, колдонуучулар жөнүндө жазуулары сакталат. Колдонулган эстин чондугу, процессордун убактысы, катуу дисктин колдонуусу жазылат. Прокси сервердин бул окуялар үчүн бир нече лог файлдын түрдөлү бар. Кээбирлерин колдонбой койсо болот, ал эми кээбирлери болсо абдан маанилү жана кошумча тейлөөнү талап кылат.

Лог файлдардын тизмеси:

• Squid.out – эгерде прокси сервер RunScript аркылуу иштетилсе анда бул жерде серведин иштөөгө баштаган убактысы сакталат жана баардык критикалык каталар жазылат (Assert функциясы жараткан). Болбосо бул файл таза бойдон калат.

• Cache.log – прокси сервердин дебаг каталары жазылат. Көбүнчө учурда бул файл маанилү саналат, жана прокси сервердин жаңы мүмкүнчүлүгүн тестирлөөдө, тейлөөдө чоң кызыктырат.

• Useragent.log – колдонуучулардын браузерлерин, алардын версиялары жазылат.

• Store.log – кэште сакталган обьектилердин үстүндө жүргүзүлгөн кадамдар жазылат. Обьект колдонулбайт, обьект кэштен өчүрүлдү, обьект диске сакталды, обьект дискте сакталган жана кайрадан окулган - сыяктуу.

• Access.log – эң маанилүү лог файл. Баардык лог анализаторлор бул файлдын жазууларына таянып рапорторду, статистикаларды жаратат. Бул файлдын структурасына өзгөчө маани бурабыз, төмөндө деталдуу түрдө каралат. Ар бир суроо-талап транзакциясынын негизинде файл орточо 100-200 байт ылдамдыгы менен өсөт.

1.4 Access.log лог файлы

Бул файлда абдан терең жана бай белгилер жазылат. Ар бир жазуу төмөнкү форматка таянат: "%9d.%03d %6d %s %s/%03d %d %s %s %s %s%s/%s %s". Бир сап 10 өтүнүчтүү эмес мамычадан турат. Мамычалар бош жер менен бөлүнөт.

```
    1066037222.011   19120 12.83.179.11 TCP_MISS/200 359 GET
http://ads.x10.com/720x300/Z2FtZ3JlZXRpbmcxLmRhd/7/AMG
DIRECT/63.211.210.20 text/html

    1066037222.011   34173 166.181.33.71 TCP_MISS/200 559 GET
http://coursesites.blackboard.com:8081/service/collab/../10107064481
90/ -DIRECT/216.200.107.101 application/octet-stream

    1066037222.011   19287 41.51.105.27 TCP_REFRESH_MISS/200 500
GET http://fn.yam.com/include/tsemark/show.js DIRECT/210.59.224.59
application/x-javascript
```

Мамычалардын касиеты төмөнкүдөй:

Time- Unix системасында колдонулган убакыт өлчөмү. UTC-Coordinated Universal Time секундасында берилет, чекиттен кийин миллисекунда тактыгына чейин көрсөтүлөт. Бул убакыт оңой эле биз көнгөн убакыт өлчөмүнө которулат. Мисал: «1066037222.011»

Duration – Ар бир http суроо-талаптын алган убактысы, миллисекунда өлчөмүндө. Суроо-талаптын иштетүү башынан ийгиликтүү аяктаганга чейинки убакытты көрсөтөт. Мисал: «19120»

Client address – колдонуучунун IP адреси. Башкача айтканда ички тармактагы суроо-талап жүргүзгөн колдонуучунун адреси. Мисал «192.168.20.127»

Result codes – бул мамыча суроо-талаптын жыйынтыгын белгилейт. Эки бөлүктөн турат, слэш символу менен ажыратылат. Биринчи бөлүк Squid тин өзүнүн коду, экинчи бөлүк болсо HTTP протоколунда жарыяланган абал кодтон турат. Мисал: «TCP_REFRESH_HIT/200». Төмөнкү таблица-1'де маанилүү ачкыч сөздөр көрсөтүлгөн.

Таблица-1 SQUID прокси серверге келген суроо-талаптын жыйынтыгынын абалын көрсөткөн чоңдуктар

Код	Кыскача мааниси
TCP_HIT	Обьект кэште сакталган жана табылды
TCP_MISS	Обьект кэште табылган жок
TCP_REFRESH_HIT	Обьект кэште, бирок эскирди, жаңы суроо талаптын негизинде http 304 коду келди.
TCP_REF_FAIL_HIT	Обьект кэште, бирок эскирди, суроо талап уратылды, эски обьект клиентке жеткирилди.
TCP_REFRESH_MISS	Обьект кэште, бирок эскирди, суроо талап ийгиликтүү аяктады, жаңы обьект клиентке жеткирилди.
TCP_CLIENT_REFRESH_MISS	No-cache башатынын негизинде жаңы суроо талап жүргүзүлдү.
TCP_IMS_HIT	IMS суроо талап негизинде кайрадан обьект жүктөө буйругу жүргүзүлдү
TCP_SWATFAIL_MISS	Обьект кэште сакталган, бирок окууга мүмкүнсүз.
TCP_NEGATIVE_HIT	Кэштелбеген обьектке суроо талап. (404 катасы)
TCP_MEM_HIT	Обьект кэште сакталган жана

	оперативдик эсте.
TCP_DENIED	Бул обьектке тыюу салынган
TCP_OFFLINE_HIT	Offline модто обьект кэште сакталган жана табылды
UDP_HIT	Обьект кэште сакталган жана табылды
UDP_MISS	Обьект кэште табылган жок
UDP_DENIED	Бул обьектке тыюу салынган
UDP_INVALID	Туура эмес суроо талап.
NONE	Ката келип чыкты.

Bytes – суроо-талаптын негизинде келген обьектин өлчөмү, байт менен көрсөтүлөт. Бул жерде обьектин баш аты (header) да эсептелет. Жана суроо-талаптын негизинде каталуу жооп келиши мүмкүн, бул каталуу жоптун өлчөмү дагы саналат. Мисалы: «68853».

Request method – суроо талаптын методун көргөзөт. Мисалы «GET», «POST». Таблица-2'де баардык методтордун тизмеси берилет.

Таблица-2 SQUID прокси сервердин суроо-талапта колдонгон методтору

Метод	Жарыяланган	Кыскача түшүнүк
GET	HTTP/0.9	обьектин суроо талабы.
HEAD	HTTP/1.0	метадатанын суроо талабы.
POST	HTTP/1.0	берилиш жиберүү
PUT	HTTP/1.1	берилиш жүктөө
DELETE	HTTP/1.1	берилишти өчүрүү
TRACE	HTTP/1.1	суроо талапты андуу
OPTIONS	HTTP/1.1	суроо талаптын касиети.
CONNECT	HTTP/1.1r3	SSL байланыш.
ICP_QUERY	Squid	ICP алмашуу.
PURGE	Squid	кэштен обьекти өчүрүү.
PROPFIND	rfc2518	обьектин касиетин суроо.
PROPATCH	rfc2518	обьктин касиетин өзгөртүү

MKCOL	rfc2518	жаңы байланышш түзүү.
COPY	rfc2518	обьектин көчүрмөсүн түзүү.
MOVE	rfc2518	обьекти ташуу
LOCK	rfc2518	обьекти өзгөртүүдөн коргоо.
UNLOCK	rfc2518	обьектке өзгөртүү уруксаты.

Cache.log файлы

Бул файлда сервердин иштөө учурунда пайда болгон окуялар сакталат. Каталар, эскертүүлөр, маалымат билдирүүлөр келтирилет. Бул файл абдан маанилүү файл, жана сервердин иштөөсүндө ката пайда болгондо эң биринчи орунда каралуучу жер болот. Файлдын ичинде мисал катары төмөнкү саптарды көрүүгө болот:

```
2010/09/29 12:09:45| Starting Squid Cache version 2.5.STABLE4 for i386-freebsd4.8...

2010/09/29 12:09:45| Process ID 18990

2010/09/29 12:09:45| With 1064 file descriptors available

2010/09/29 12:09:45| Performing DNS Tests...

2010/09/29 12:09:45| Successful DNS name lookup tests...

2010/09/29 12:09:45| DNS Socket created at 0.0.0.0, port 1154, FD 5

2010/09/29 12:09:45| Adding nameserver 24.221.192.5 from /etc/resolv.conf

2010/09/29 12:09:45| Adding nameserver 24.221.208.5 from /etc/resolv.conf

2010/09/29 12:09:45| helperOpenServers: Starting 5 'redirector.pl' processes

2010/09/29 12:09:45| Unlinkd pipe opened on FD 15

2010/09/29 12:09:45| Swap maxSize 10240 KB, estimated 787 objects

2010/09/29 12:09:45| Target number of buckets: 39

2010/09/29 12:09:45| Using 8192 Store buckets

2010/09/29 12:09:45| Max Mem  size: 8192 KB

2010/09/29 12:09:45| Max Swap size: 10240 KB

2010/09/29 12:09:45| Rebuilding storage in /usr/local/squid/var/cache (CLEAN)
```

```
2010/09/29 12:09:45| Using Least Load store dir selection

2010/09/29 12:09:45| Set Current Directory to
/usr/local/squid/var/cache

2010/09/29 12:09:45| Loaded Icons.

2010/09/29 12:09:45| Accepting HTTP connections at 0.0.0.0, port
3128, FD 16.

2010/09/29 12:09:45| Accepting ICP messages at 0.0.0.0, port 3130,
FD 17.

2010/09/29 12:09:45| WCCP Disabled.

2010/09/29 12:09:45| Ready to serve requests.
```

Ар бир сап убакыт өлчөмүндөн башталат, мындайча ар бир билдирүүнүн жаралган убактысын көрүүгө болот. Жогорудагы мисалда cache.log файлынын башкы гана саптары келтирилиген, жана бул жерде сервердин ийгиликтүү башталышын, жүктөлүүсүн жана колдонуучуларга сервис бере алатурган абалда экендигин көрүүгө болот. Ошондой эле каталарда бул жерде көрсөтүлөт.

Cache.log файлы негизи абдан жай өсүүчү файл болуп саналат, бирок кадиски эмес http транзакция негизинде же болбосо DDOS атака негизинде же болбосо кандайдыр вирустун иштөөсүнөн бул файл абдан бат өсүшү белгилүү. Ошондуктан бул файл дагы лог файлдардын ротациясы алдына алынышы керек. Мындайча биз катуу дисктин кокусунан толуп кетишинен сакталып калабыз.

Дебаг дэңгээл 1- ден чоң болгондо бул файл көптөгөн деталдуу эскертүүлөр жана маалыматтарды камтыйт, жана кийинки учурларда кандайдыр бир окуяны териштирүүдө кыйынчылык келтириши мүмкүн, ошондуктан дебаг дэңгээл 1-ден жогоруу болгондо абдан абайлоо керек. Ошондой эле лог анализатордун иштөөсүнө жана сервердин өзүнөн абдан чоң рессурсту талап кылат.

Cache.log файлына жазылган билдирүүлөрдү администратордун терминалына да жиберүү мүмкүн. Анын үчүн

```
/usr/local/squid/sbin/squid -d1
```

командасын колдонуу керек.

Store.log файлы

Бул файлда серведин кандайдыр бир обьекттин кэште сактоосун, же сактообосун чечүүчү билдирүүлөр сакталат. Ар бир сакталуучу, өчүрүлүүчү, сакталбоочу обьектилер үчүн бир саптык маалымат жазылат. Ал бетте бул маалымат абдан төмөнкү дэңгээлдэги маалыматты камтыйт, жана күнүмдүк жашоодо аз колдонулат. Бирок биздин изилдөөбүздүн максатына жетишүү үчүн бул файл абдан чоң ролду ойноору шексиз. Store.log файлы текстик, жана төмөнкүдөй саптарды камтыйт:

```
1067299212.671 RELEASE -1 FFFFFFFF 5ECD93934257594825659B596D9444BC
200 1067299023 1034873897 1067299023 image/jpeg 3386/3386 GET
http://ebiz0.ipixmedia.com/abc/ebiz/_EBIZ_3922eabf57d44e2a4c3e7cd234
a...

1067299212.786 RELEASE -1 FFFFFFFF B388F7B766B307ADEC044A4099946A21
200 1067297755 -1 -1 text/html -1/566 GET
http://www.evenflowrocks.com/pages/100303pic15.cfm
```

Бул жерде

Timestamp – окуянын жаралган убактысы (1067299212.671)

Action – обьектин үстүндө жүргүзүлгөн операция. Үч маанини алышы мүмкүн: Swapout (обьект ийгиликтүү кэшке сакталды), Release(обьект ийгиликтүү кэштен өчүрүлдү), So_Fail (обьектти кэшке сактоодо операция каталуу аяктады).

Directory number – папканын ондук сан менен жазылган аты (белгилүү болгондой SQUID сервердин кэш папкалары номер менен аталат). Эгерде обьект папкада жайгашпаса анда (-1) деген мааниге ээ болот.

File number – 25биттик файл идентификатор, бул идентификатор жардамы менен кэште сакталган обьекттин толук жолун табуу мүмкүн. Release жана Swapout операциясында бул чоңдук FFFFFF маанисине ээ болот.

Cache key – md5 хэш ачкыч, бул ачкыч жардамы менен сервер обьекттин сакталуу локациясын таба алат.

Status code – http суроо талаптын аяктоосунда пайда болгон статус код

Date – http жооптун анык убактысы, (-1) маани http жооптун каталуу анык убакытысы менен келгендигин билдирет, ал эми (-2) маниси болсо таптакыр бул чоңдук http жоопто жок келгендигин билдирет.

Last modified – атайын ачкычтык «last modified» чоңдугунун мааниси жана обьектин эң акыркы жолу өзгөртүлгөн убактысын көрсөтөт.

Expires – обьектин эскирүү убактысын көрсөтөт.

Content type - атайын ачкычтык «content type» чоңдугунун мааниси жана обьектин түрүн билдирет.

Content length – обьктин байт өлчөмүндөгү көлөмүн көрсөтөт.

Method – http протоколунда колдонгон метод түрүн көрсөтөт.

URI – сакталуучу обьектинин оригиналдуу жайгашкан толук адреси.

Серверди жүктөө

SQUID прокси сервер көбүнчө Линукс операциондук системалар үйбүлөсүндө кошо орнотулуп келет. Эгерде өзүнчө орнотуу керек болсо анда түздөн түз эле официалдык сайттан жүктөп алуу болот. Бул учурда администраторлор көптөгөн функцияларды кошуп же алып салуу мүмкүнчүлүгүнө ээ болушат. Же болбосо «yum» утилита жардамы менен ортонуу мүмкүн (Fedora Core, RedHat). Бирок бул учурда бул утилиталар колдонуучудан көп сурөө сурабай эле программдык жабдууну корфигурациялап орнотушат.

Мисалы: yum install Squid

же жаңылатуу керек болсо:

```
yum update Squid
```

SQUID орнотулгандан кийн аны иштетүү үчүн:

```
service squid start
```

командасы колдонулат. (ошондой эле кайра жүктөө командасы «service squid restart» же токтотуу командасы «service squid stop»)

SQUID баардык каталарды /var/log/squid/output.log файлында сактайт. Эгерде кандайдыр бир каталар орун алса, же SQUID прокси сервери туруксуз иштей баштаса биринчиден бул файлга кайрылыш керек.

SQUID прокси сервердин баардык настройкалар /etc/squid/squid.conf файлында сакталат. Баардык конфигурациялык өзгөрмөлөр бул файлдын ичинде англис тилинде жазылган. Бул файлдын эң маанилүү бөлүктөрүн карап кетсек:

Параметр http_port бул параметр бир нече түрдө жазылышы мүмкүн.

```
Хосттун_аты: порт
IP_адрес : порт
```

Бул жерде порт, прокси сервердин иштөө порту. Бул порт аркылуу прокси сервер суроо талаптарды колдонуучулардан алат. HTTP прокси серверлер көбүнчө учурда 3128 портунда иштешет. Хосттун аты бул кайсы тармак интерфесте иштөөсүн көрсөтөт. Прокси сервер бир нече тармак интерфейстер менен иштей алат. Ошондо реалдуу мисал келтирсек төмөнкүдөй маанилер алынышы мүмкүн:

```
http_port 192.168.20.1:3128
http_port eth1:3128
```

Параметр https_port бул параметр жогорку параметрге окшош, жөн гана https протоколунун портун белгилейт.

Паметр icp_port бул параметр SQUIDпрокси серверин башка прокси серверлер менен кэш суроо талаптарды жүргүзүүчү портун белгилейт. Эгерде биздин прокси сервер кэшти башка прокси серверлер менен бөлүштүрбөсө бул параметрди өчүрүп коюболот.

Параметр cache_mem эң маанилүү параметрледин бири. SQUID ке бөлүнүп берилген коп колдонулган обьекттердин оперативдик эстеги чоңдугүн аныктайт.

Параметр cache_swap_low, cache_swap_high дискти максималдуу (процент менен берилет) колдонуу параметрлери.

1.5 Squid'тин командалык сап опциялары

Squid прокси сервери командалык саптын жардамы менен иштетилет, жана GUI интерфейси жок болот. Ошондуктан прокси сервери командалык сапта көптөгөн опцияларды камтыйт.

-a port: Жаңы http_port маанисин берет. Бул опция дайыма squid.conf конфигурациялык файлдагы http_port маанисинен жогору болот.

-d level: Прокси сервер дебаг билдирүүлөрдү stderr го жазат. (ошондой эле cache.log файлына). Бул жерде level чоңдугу дебаг билдирүүнүн дэңгээлин билдирет. Коп учурда "-d1" жетиштүү жана информативдүү болот.

-file: Альтернативдүү конфигурациондук файлдын жолун көрсөтөт.

-h: Прокси серведин жардамчы баракчасын көрсөтөт жана ар бир опцияга кыскыча баяндама берилет

-k function: Прокси серверге атайын сигналдарды жиберүү опциясы. Бул жерде "function" төмөнкүлөрдөн болушу мүмкүн: reconfigure, rotate, shutdown, interrupt, kill, debug, check, parse. Ар бир опция прокси сервердин иштөөсүнө жана абалына административдик башкаруу мүмкүнчүлүгүн берет.

-v: Прокси сервердин версиясын көрсөтөт

-z: Cache папкаларды жаратуу үчүн колдонулат. Прокси сервер эң биринчи жолу иштегенде бул опция дайыма колдонуу зарыл. Болбосо прокси сервер каталуу аякталат.

-D: DNS тестирирлөөнү өчүрөт. Башкача айтканда прокси сервер ДНС сервердин иштөөсүн текшербеден баштатылат. Бул опция колдонбосо прокси сервер ДНС серверди текшерет, жана ДНС жооп келбесе прокси сервер баштатылбайт.

-F: Прокси серверди баардык суроо талаптарды кабыл албоо абалына өткөрөт.

-N: Прокси сервердин «демон» процессине өткөрбөө опциясы.

-X: Толук дебаг абалына алып келет.

1.6 Named Pipes түшүнүгү

Pipe-англис тилинде «түтүк» дегенди билдирет. Бул термин Unix операциндук системасындан келет. Бул операциондук системаларда named pipe тын жардамы менен эки ар түрдүү процесстердин бири-бири менен маалымат алмашууна жардам берет. Named pipe ты бир процесс окуйт, ал эми экинчи процесс named pipe ка жазат. Башкача айтканда түтүктүн бир башында жазуучу процесс ал эми экинчи башында окуучу процесс жайгашкан. Мындан named pipe тын бир гана тарапка (half – duplex) маалымат өткөрө алатурганы түшүнүктүү. Эгерде эки тарапка маалымат жиберүү керек болсо (full – duplex) анда эки named pipe жаратуу керек, жана ар бир процесс бир named pipe ты окуу үчүн ал эми экинчи named pipe ты жазуу үчүн ачыш керек. Тескеринче экинчи процесс болсо биринчи named pipe ты жазуу ал эми экинчи named pipe ты окуу үчүн ачыш керек. Ошондо эки ар түрдүү процесстер арасында full – duplex маалымат агышы түзүлөт. Named pipe ты окуучу процесс болмоюнча жазуучу процесс блоктолот. Башкача айтканда named pipe тын иштешүү үчүн окуучу жана жазуучу процесстер болушу зарыл. Болбосо процесстер токтолуат (блоктолот). Ошондуктан named pipe ты колдонууда абдан чоң ишмердүүлүктү талап кылат. [2]

Named pipe ты жаратуу

named pipe эки жол менен жаратылышы мүмкүн. Бул командалык саптын (comman line) жардамы менен же болбосо программдык жол менен жаратуу мүмкүн. Командалык сапта «mkfifo» жана «mknod» жардамы менен түзүлөт.

Мисал:

```
mkfifo /tmp/pipename.log
```

Ал эми программа жолу менен болсо төмөнкү функция чакырылыш керек.

```
int mkfifo (const char* path, mode_t mode)
```

named pipe окуу үчүн ачуу open() же fopen() функциялары чакырылыш керек.Эгерде функция катасыз иштеп бутсө ал кайрадан файл дискрипторду кайтарып берет. Жана бул файл дискриптордун жардамы менен жөнөкөй жазуу (write) окуу (read) операцияларын аткаруу болот. [2]

named pipe бир учурда окуу жана жазуу үчүн ачылбашы керек. Процесс же окуу же болбосо жазуу модунда named pipe ты ачуу зарыл. Бир түрдөн (окуу же жазуу) ачылган named pipe кайрадан башка түргө өтө албайт, жабылышы зарыл. [2]

named pipe тын өзгөчөлүктөрү

- named pipe колдонууда оңой жана жөнөкөй
- эч кандай синхронизацияны керектирбейт
- жазуу операциясы «атомдук» болуп саналат. Башкача айтканда «заматта» жазуу операциясы аткарылат.
- named pipe тын жөнөкөй файлдардыкындай коопсуздук касиеттери бар

named pipe тын кемчиликтери

- named pipe бир компьютерде жаратылган процесстин арасында гана колдонула алат. Башкача айтканда тармак аркылуу маалымат алмашууда колдонулбайт.
- named pipe локалдык файл системада жаратылат. Ошондуктан named pipe ты биз NFS файл системасында жарата албайбыз.
- Программаладоодо абдан чоң кылдаттыкты талап кылат (блокировкаларды жаратпоо үчүн).

БӨЛҮМ – 2 АНАЛИЗ ЖАНА МЕТОДТОР

2.1 Учурда бар болгон анализаторлор

Төмөнкү таблица-3'тө SQUID тин жараткан лог файлын анализдөөчү программалардын тизмеси жана кыскача өзгөчөлүктөрү көрсөтүлгөн.

Таблица-3 Учурда бар болгон лог-анализаторлордун тизмеси жана касиеттери

Аты	БД	Прог.тили	ОС	Шилтеме
Free SA	**Жок**	C/PHP	Linux, Unix	http://sourceforge.net/projects/free-sa/files/
LightSquid	**Жок**	Perl	Linux, Unix	http://lightsquid.sourceforge.net/
ProxyStat	**Жок**	Perl	Linux, Unix	-
MySAR	MySQL	PHP/C++	Linux, Unix	http:// giannis.stoilis.gr/software/mysar/
SAMS	MySQL	C++	Linux, Unix	http://sams.perm.ru/
SARG	**Жок**	C	Linux, Unix	http://sarg.sourceforge.net/
Squid Traffic Counter	**Жок**	Sh, Perl, CGI	Linux, Unix	http://stc.nixdev.org
Squid2MySQL	MySQL	PHP	Linux, Unix	http://evc.fromru.com/squid2mysql/index.html
SquidGuard	**Жок**	C	Linux, Unix	http://www.squidguard.org/
Squid-PB	MySQL	PHP	Linux, Unix	http://pb.pils.ru/
Statman	PostreSQL	Perl	Linux, Unix	http://cyberos.narod.ru/statman/index.html

FreeSA- жаңы лог анализатор болуп саналат, жана C тилинде жазылган. Функционалдык жактан абдан SARG жана LightSquid анализаторлорго окшош.

Эң негизги айырмачылык бул отчеттун жаратуу убактысы 7 жана 20 эсе бат. Кошумча серверди баалоо жана сыноочу отчетторду камтыйт. Журналдык файлдардын ар түрдүү форматтарын тейлей алат, алардын ичине SQUID, CLF, Postfix, Qmail, CGP жараткан лог файлдарды анализдейт. Отчеттор html файлдарда сакталат. [4]

LightSquid – бат жана жыйнак SQUID лог анализатор. Perl тилинде жазылган база катары SARG колдонулган. Оңой жана бат жүктөлөт, толукча модульдарды талап кылбайт. SARG'ка салыштырмалуу катуу диске аз орун ээлейт. Статистикаларды кароо үчүн веб интерфейси бар. Отчеттер файл түрүндө сакталат. Жалпы отчет, колдонуучулар боюнча отчет, ар күндөр боюнча отчет, убакыт боюнча отчет жаратат. [6]

Squid-PB – эң маанилүү мүмкүнчүлүк бул колдонуучуларды лимит аяктаганда интернетке кирүүгө тыйюу салуу. C- тилинде жазылган, жана веб интерфеси бар. Берилиштер базасын колдот PHP+MySQL. [12]

Squid Traffic Counter – бир нече скриптердин тизмесинден турган система (sh, perl, perl+CGI) колдонуучуларды аутентификация жана лимиттөөчү мүмкүнчүлүктөрүн камтыйт. Веб интерфеси бар, жана узактан администрлөөгө мүмкүндүк берет. [9]

Squserlim- колдонуучуларга квота аркылуу интернетке кирүү мүмкүнчүлүгү бар, квоталар каалаган мөөнөткө жарыяланышы мүмкүн. Квоталар MySQL де сакталат.

SquidParser – колдонуучуларды башкаруу системасы, баланстын учетун жүргүзөт, колдонуучуларды оңой тейлөө мүмкүнчүлүгү бар. Perl жана MySQL менен жазылган.

SAMS- абдан ыңгайлуу жана оңой тейлөө мүмкүнчүлүн камтыйт. Прокси серверге колдонуучуларды аутентификация аркылуу уруксат берет. Жана Windows системалардагы NTLM аутентификация протоколун. [5]

2.2 Эң популярдуу анализаторлорду изилдөө

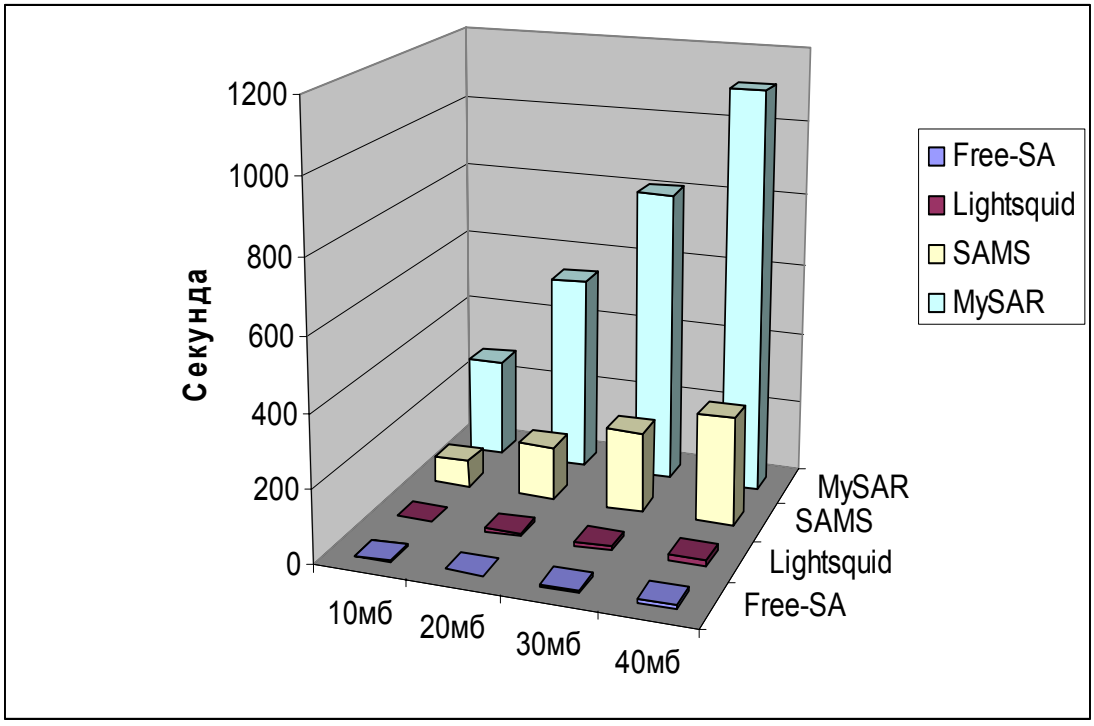
Бул жерде эң популярдуу жана кеңири таралган анализаторлор булар LightSquid, FreeSA, SAMS, MySAR. Булардын баардыгы «GNU General Public

License»[7] лицензиясы менен эркин таркатылат. Бул үч анализаторлор бир системага[8] орнотулуп, жана иштөө убактысы боюнча салыштырылып төмөнкү таблица-4төгү жыйынтыка жетиштик.

Таблица-4 Лог-анализаторлордун тест жыйынтыктары

	Access.log файлдын көлөмү			
	10Mb	20Mb	30Mb	40Mb
Free-SA	2,21 сек	4,79 сек	7,03 сек	10,72 сек
Lightsquid	3,6 сек	7,61 сек	10,76 сек	14,73 сек
SAMS	73,18 сек	148,81 сек	223,43 сек	298,45 сек
MySAR	273,57сек	538,17 сек	808,36 сек	1113.11 сек

График-1 Таблица - 4 түн негизинде курулган график.



[7] http://www.gnu.org/copyleft/gpl.html

[8] FreeBSD 8.0 RELEASE, Squid Cache 2.7 STABLE7, MySQL 5.0.86, Apache 2.2.13, GCC 4.2.1, PHP 5.2.11, Perl 5.8.9

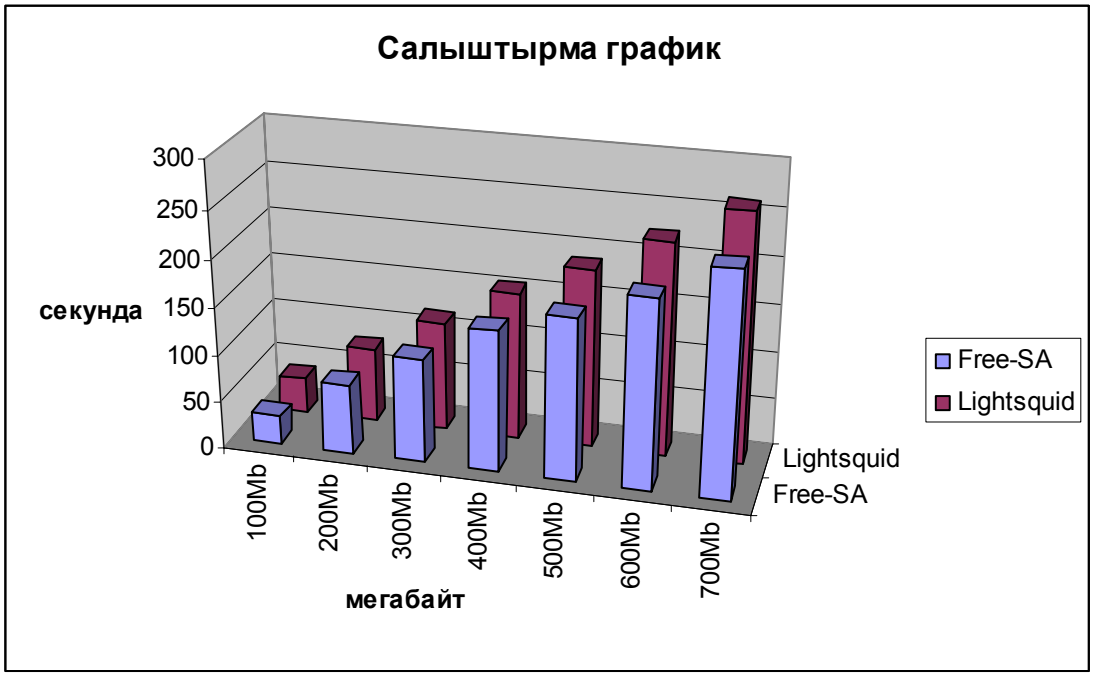
Бул жерде көрүнгөндөй SAMS жана MySAR абдан жай иштөөдө. Анын себеби SAMS жана MySAR баардык маалыматты берилиштер базасында сакташат. Ошондуктан лог файлды анализдөөдө Free-SA жана Lightsquid тен абдан артта калышкан. Бирок БД-да сакталган маалымат үстүндө кошумча амал жүргүзүү мүмкүн. Ошондой эле SAMS тын абдан ыңгайлуу Web-интерфейси бар. Ушул өзгөчүлүгү үчүн SAMS популярдуу болуп келе жатат.

SAMS жана MySAR иштөө убакыты боюнча абдан артта калган үчүн кийнки тестерде катышка алынган жок. Ал эми төмөнкү таблица-5 боюнча Free-SA анализатору access.log файлы чоң болгон сайын Lightsquid-тен батыраак иштеши көрүнүп турат.

Таблица-5 Лог-анализаторлордун тест жыйынтыктары (access.log файлынын көлөмү өзгөрүлгөн)

	Access.log файлдын көлөмү						
	100Mb	200Mb	300Mb	400Mb	500Mb	600Mb	700Mb
Free-SA	30,6 сек	73,37 сек	109,17 сек	147,91 сек	169,75 сек	197,3 сек	233,68 сек
Lightsquid	34,73 сек	77,96 сек	115,33 сек	155,29 сек	188,31 сек	224,45 сек	262,33 сек
айырма	**4,13**	**4,59**	**6,16**	**7,38**	**18,56**	**27,15**	**28,65**

График - 2. Таблица - 4 түн негизинде курулган график



Бул кубулуш Free-SA анализатору «C» тилинде, ал эми Lightsquid болсо «Perl» тилинде жазылгандыгы үчүн байкалууда.

Изилдөөнүн негизинде бүгүнкү күндө бар болгон жана кеңири колдонулган Squid прокси сервердин анализаторлордун ичинен эң бат иштеген FreeSA болуп чыкты. Ал эми эң ыңгайлуу жана оңой тейлөө өзгөчүлүгү менен SAMS жеңип чыкты.

Access.log файлы чоң болгон сайын БД'ны колдонгон анализаторлор абдан ыңгайсыз болоору көрүнүктүү (SAMS, MySAR). Бул ыңгайсыздыкты cron утилитасы жардамы менен чечүү мүмкүн (мисалы: анализатор ар 5 минута сайын иштетилет, мындайча access.log файлдын көлөмү кичине болуп турат).

Бирок эч бири жогоруда көрсөтүлүп кеткен проблемаларды толук түрдө чече албайт. Free-SA бат иштегени менен төмөнкү кемчиликтери бар:

жыйынтык/отчет файл түрүндө сакталат;

онлайн мониторинг өзгөчөлүгү жок (cron жардамы керек).

SAMS болсо

абдан жай иштейт;

онлайн мониторинг өзгөчүлүгү 10сек интервал менен иштейт

2.3 The Cache Manager

SQUID прокси серведин ички компоненттеринин статусун көрүү үчүн «cache manager» деген протокол-интерфейси бар. Бул интерфейске жөнөкөй эле HTTP суроо талап менен кирүү мүмкүн. Толук HTTP суроо талап төмөнкүдөй болот:

```
cache_object://cache.host.name/info
```

«cache manager» интерфесиндеги маалыматты squidclient утилитасы жардамы менен көрүү болот. Эң жөнөкөй түрдө командалык сапта иштейт жана командалык сапка маалыматты чыгарат. Бул кээбир чоң маалыматты, таблицаларды көрүүдө/окууда кыйнчылык жаратат. Squidclient cache_manager'деги маалыматты файлга сактап калып, кийин анализдөө үчүн колдонушу мүмкүн.

Cache_manager'деги маалымат абдан маанилүү болот, жана башка колдонуучулар тарабынан көрүнбөш керек. Ошондуктан squid.conf файлында атайын ишенимдүү IP адрестердин тизмесин киргизип жана тизмеден сырткары эч кимге уруксат бербөө керек.

2.4 Cache_manager'дин бөлүмдөрү

Төмөнкү таблица-6'да cache_manager'дин бөлүмтөрү жана алардын кыскача түшүнүктөрү келтирилген. Жылдызча менен белгиленген бөлүмдөр сыр сөздү (паролду) талап кылат.

Таблица-6 cache manager протоколунун бөлүмдөрү

Бөлүм аты	**Кыскача түшүнүк**
leaks	Memory Leak Tracking
mem	Memory Utilization
cbdata	Callback Data Registry Contents
events	Event Queue
squidaio_counts	Async IO Function Counters
diskd	DISKD Stats
config	Current Squid Configuration*
comm_incoming	comm_incoming( ) Stats
ipcache	IP Cache Stats and Contents
fqdncache	FQDN Cache Stats and Contents
idns	Internal DNS Statistics
dns	Dnsserver Statistics
redirector	URL Redirector Stats
basicauthenticator	Basic User Authenticator Stats
digestauthenticator	Digest User Authenticator Stats
ntlmauthenticator	NTLM User Authenticator Stats
external_acl	External ACL Stats
http_headers	HTTP Header Statistics

Бөлүм аты	Кыскача түшүнүк
via_headers	Via Request Headers
forw_headers	X-Forwarded-For Request Headers
menu	This Cache Manager Menu
shutdown	Shut Down the Squid Process*
offline_toggle	Toggle offline_mode Setting*
info	General Runtime Information
filedescriptors	Process File Descriptor Allocation
objects	All Cache Objects
vm_objects	In-Memory and In-Transit Objects
openfd_objects	Objects with Swapout Files Open
io	Server-Side Network read( ) Size Histograms
counters	Traffic and Resource Counters
peer_select	Peer Selection Algorithms
digest_stats	Cache Digest and ICP Blob
5min	5 Minute Average of Counters
60min	60 Minute Average of Counters
utilization	Cache Utilization
histograms	Full Histogram Counts
active_requests	Client-Side Active Requests
store_digest	Store Digest
storedir	Store Directory Stats
store_check_cachable_stats	storeCheckCachable( ) Stats
store_io	Store IO Interface Stats
pconn	Persistent Connection Utilization Histograms
refresh	Refresh Algorithm Statistics
delay	Delay Pool Levels
forward	Request Forwarding Statistics
client_list	Cache Client List

Бөлүм аты	Кыскача түшүнүк
netdb	Network Measurement Database
asndb	AS Number Database
carp	CARP Information
server_list	Peer Cache Statistics
non_peers	List of Unknown Sites Sending ICP Messages

Leaks: Memory leak tracking

Бул бөлүмдөгү маалымат SQUID прокси сервер ./configure –enable-leakfinder опциясы менен компиляцияланганда гана көрсөтүлөт. Жана компьютердик эстин туура эмес колдонуусун, жоголуусун көрсөтөт.

Mem: Memory Utilization

Бул бөлүмдө комьютердик эстин колдонуусунун таблицалык түрдө маалымат көрсөтөт. Ар бир саптагы маалымат атайын бөлүнгөн эстеги көптүкө таандык болот. Ар бир эс көптүктүн ичинде бош эстин көлөмү, эс көптүктүгүнүн толугу менен толушунун убактысы, эс көптүгүнүн бошотулушу.

Cbdata: Callback data registry contents

Бөлүм SQUID прокси сервердин ички көрсөткүчтөрдүн (pointers) башкаруусу, колдонуусу жөнүндө маалымат камтыйт.

Events: Event queue

SQUID прокси сервер колдонуучулардын суроо-талабы менен бир убакытта аткарыла турган event’терди атайын queue-катарда сактайт. Мисалы кэш баракларды бошотуу, бул event ар бир секунда сайын болуп турат. Мындайча прокси сервер учурда аткарып жаткан event’терди көрүүгө болот.

Squidaio_counts: Async IO function counters

Бул бөлүмдөгү маалымат SQUID прокси сервер ./configure –enable-sterio=aufs опциясы менен компиляцияланганда көрүнөт. Учурда ачык, жабык, аткарылып жаткан, токтолуп жаткан суроо талаптарды көрүүгө болот.

Diskd: Diskd stats

Бул бөлүмдөгү маалымат SQUID прокси сервер ./configure – enable-sterio=diskd опциясы менен компиляцияланганда көрүнөт. Катуу дискти башкаруучу diskd – аттуу демонго кайрылуу статистикасы көрсөтүлөт. Жазуу, окуу, жаратуу, өчүрүү талабынын саны, ар бир талабтын ката менен аяктоо саны.

Config: Current squid configuration

Бул бөлүмдө прокси сервердин конфигурациялык файлынын камтыган маалыматын (squid.conf) көрүүгө болот. Бул бөлүм кошумча сыр-сөз менен коопсуздук көз караштан сакталган. Squid.conf файлында cachemgr_passwd опцисы менен сыр-сөз көрсөтүлүш керек. Кокустан squid.conf файлы өчүрүлүп же болбосо туура эмес өзгөртүүлөр киргизилсе бул жерден кайрадан эски конфигурациялык файлды окуп алуу мүмкүн (албетте иштеп жаткан прокси серверге өзгөртүүлөрдү кайрадан окуу буйругу берилбесе)

Comm._incoming: comm._incomming() stats

Бөлүмдө прокси сервердин комьютердик тармагында окуу-жазуу операциясынын статистикасы көрсөтүлөт. HTTP суроо талаптын келүү сокетинин текшерүү отчету. Колдонуучулардан суроо талап ушул сокеттен келгени үчүн бул сокет абдан маанилүү жана ар дайым көп рессурсту талап кылат. Жана прокси серведин иштөөсүнө таасир берүүчү көрсөткүч болуп саналат.

IPcache: IP cache stat and contents

Хосттун аты жана анын айпи адресинин маалыматы көрсөтүлөт (dns жазуулар). Мисалы төмөнкүдөй:

```
IPcache Entries: 10034
```

```
IPcache Requests: 1066445

IPcache Hits: 817880

IPcache Negative Hits: 6846

IPcache Misses: 200497
```

Бул мисалда 10034 хост жөнүндө жазуу бар болдугу, прокси серверге 1066445 суроо талап келгени алардын ичинен 817880 суроо талапта кэште хост жөнүндө маалымат табылганы көрсөтүлөн.

Жана ар бир хост жөнүндө маалымат төмөнкү мисалдай көрсөтүлөт: Хосттун аты, хостун акыркы жолу колдонуу убактысы, хостун эскирүү убактысы, хосттко таандык айпи адрестердин саны, айпи адресстердин тизмеси жана статусу.

```
Hostname                        Flg lstref    TTL  N

 ads.x10.com                              9    110  1( 0)
63.211.210.20-OK

 us.rd.yahoo.com                        640   -340  4( 0)
216.136.232.150-OK


216.136.232.147-OK


216.136.232.149-OK


216.136.232.148-OK

 www.movielodge.com                    7143  -2161  1( 0)
66.250.223.36-OK
```

fqdncache: FQDN cache stats and contents

Бул бөлүмдө жогорку бөлүмдөй маалымат сакталат бирок бир гана айырмачылык маалымат айпи адрестердин хост атына таандыгын көрсөтөт.

Idns: Internal DNS statistics

SQUID прокси серверде ички DNS клиенти бар, бул клиенттин статистикалары бул бөлүмдө көрүүгө болот:

```
Internal DNS Statistics:
The Queue:
                        DELAY SINCE

  ID   SIZE SENDS FIRST SEND LAST SEND

------ ---- ----- ---------- ---------

001876   44     1      0.010     0.010

001875   44     1      0.010     0.010
```

DNS серверден жооп келе элек суроо талаптардын тизмеси, бул тизме чоң болсо демек DNS серверде бир проблема бар болдугуну билдирет. Суроо талаптын жиберилгенден бери өткөн убакыт, жана кайталанып жиберилген днс-суроо талаптын ийгиликтүү жооп саны.

```
Nameservers:
IP ADDRESS      # QUERIES # REPLIES
--------------- --------- ---------

192.168.19.124       4889      4844

192.168.19.190         91        51

192.168.10.2           73        39
```

DNS серверлердин айпи адресстери, жана DNS серверге жиберилген днс суроо талаптардын саны, ийгиликтүү днс жооптордун саны. Прокси сервер ар дайм тизмеде биринчи турган DNS серверге суроо талап жиберет. Бириинчи серверден ийгиликтүү жооп келбегенде гана тизмеде экинчи серверге суроо талап жиберилет.

```
Rcode Matrix:
RCODE ATTEMPT1 ATTEMPT2 ATTEMPT3
```

```
0      6149         4         2
1         0         0         0
2        38        34        32
```

RCODE – 0 чоңдугу суроо талаптка ийгиликтүү жооп келгендигин көрсөтөт, RCODE-1, RCODE-2, RCODE-3, RCODE-4, RCPDE-5 болсо ар кандай каталардын пайда болушунун санын көрсөтөт. ATTEMPT1 суроо талаптын биринчи кайталоосунда ийгиликтүү жооптордун келген санын көрсөтөт.

Dns:Dnsserver statistics

Бул бөлүмдөгү маалымат SQUID прокси сервер ./configure –disable-internal-dns опциясы менен компиляцияланганда көрүнөт. Бул учурда прокси сервер сырткы dnsserver процесстерди жаратат, жана бул процесстерге днс-суроо талаптарды жөнөтөт.

```
Dnsserver Statistics:
number running: 5 of 5
requests sent: 3001
replies received: 3001
queue length: 0
avg service time: 23.10 msec
```

Number running – бир убакытта иштетиле ала турган кошумча днс-сервер программасынын саны жана иштеп жаткан кошумча днс-сервер программасынын саны.

Requests sent, request received – суроо талаптын жөнөтүлгөн саны жана анын кайра келген туура жооптордун саны.

Queue length – днс суроо талаптардын тизмесинин узундугу. Эгерде бул жерде нолдон сырткары сан көрсөтүлсө демек прокси сервердин бир убакытта иштетиле турган жардамсы днссервер программасынын санын көбөйтүү керек.

Dnsserver прокси сервердин кошумча-жардамчы программасы болуп саналат (helper). Башка кошумча-жардамчы программалар булар: редайректтерлер, аутентикаторлор. Баардык кошумча-жардамчы программалардын cache_manager'де бөлүмдөрү болот.

Redirector: URL redirector stats

Бул бөлүм редайректор колдонгондо гана маалымат камтыйт. SQUID прокси сервер үчүн көптөгөн редайректорлор жазылган, бирок алардын баары бирдей маалымат беришет жана жогорку бөлүмгө (DNS statistics) окшош болот.

Basicauthenticator: Basic user authentication stats

Бул бөлүмдөгү маалымат SQUID прокси сервер ./configure – enable-auth=basic опциясы менен компиляцияланганда жана squid.conf конфигурациялык файлад "auth_param basic program" директивасын аныктаганда көрүнөт.

Digestauthenticator: Digest user authenticator stats

Бул бөлүмдөгү маалымат SQUID прокси сервер ./configure – enable-auth=digest опциясы менен компиляцияланганда жана squid.conf конфигурациялык файлад "auth_param digest program" директивасын аныктаганда көрүнөт.

Ntlmauthenticator: NTLM user authenticator stats

Бул бөлүмдөгү маалымат SQUID прокси сервер ./configure – enable-auth=ntlm опциясы менен компиляцияланганда жана squid.conf конфигурациялык файлад "auth_param ntlm program" директивасын аныктаганда көрүнөт.

External_acl: external ACL stats

Бул бөлүмдө сырткы ACL (эреже, тартип) программасынын статистикасы көрсөтүлөт. SQUID прокси сервер үчүн көптөгөн «сырткы эреже» программалары бар. Администраторлор да өздөрүү жаза алышат.

http headers: HTTP header statistics

Бул бөлүмдө http-башаттардын статистикалары көрсөтүлөт. 4-секциядан турат: HTCP жооптордун статистикасы, HTTP суроо талап статистикасы, HTTP жооп статистикасы, HTTP field статистикасы.

Таблица-7 http баш аттар (headers)

Field type distribution values for HTTP requests			
ID	Name	Count	#/header
0	Accept	1425454	0.98
1	Accept-charset	320542	0.22
2	Accept-encoding	705985	0.45
3	Accept-language	1398545	0.92

Таблицада-7 де прокси сервердин 1425454 сандагы ACCEPT баш ат менен келген суроо талаптар көрсөтүлгөн. Колдонуучудан келген суроо талапка прокси сервер бир нече HTTP баш ат менен жаңы суроо талап түзүшү мүмкүн. Ошондуктан чыныгы колдонуучулардын суроо талап саны аз болушу мүмкүн.

Таблица-8 Cache control директивалар

Cache-Control directives distribution values for HTTP requests			
ID	Name	Count	#/cc_field
0	public	6866	0.02
1	private	69783	0.24
2	no-cache	78252	0.27
3	no-store	9878	0.03
4	no-transform	168	0.00
5	must-revalidate	10983	0.04
6	proxy-revalidate	2480	0.01
7	max-age	165034	0.56
8	s-maxage	4995	0.02
9	max-stale	0	0.00
10	only-if-cached	0	0.00
11	Other	9149	0.03

Via_headers: Via request headers

Бул бөлүмдөгү маалымат SQUID прокси сервер ./configure – enable-forwd-via-db опциясы менен компиляцияланганда көрүнөт. Биздин прокси сервер аркылуу башка прокси серверлердин суроо талаптары көрсөтүлөт. Коопсуздук окуяны териштирүү учурунда бул маалымат абдан пайдалуу болуп саналат. VIA баш атту суроо талаптар гана каралат, жөнөкөй колдонуучулар мындай суроо талап түзө алышпайт, жалаң гана прокси серверлер түзөт. Колдонуучуну бул бөлүмдөгү прокси сервелердин чынжырын кубалоо менен табуу мүмкүн. Мисал:

```
   4 1.0 proxy.firekitten.org:3128 (squid/2.5.STABLE1)

   1 1.0 xnsproxy.dyndns.org:3128 (squid/2.5.PRE3-20020125)

1751 1.0 nt04.rmtcc.cc.oh.us:3128 (Squid/2.4.STABLE6),

         1.0 tasksmart.rmtcc.cc.oh.us:3128 (Squid/2.4.STABLE7)

 137 1.0 reg3.bdg.telco.co.id:8080 (Squid/2.2.STABLE5),

         1.0 c1.telco.co.id:8080 (Squid/2.4.STABLE6),

         1.0 cache2.telco.co.id:8080 (Squid/2.4.STABLE1)

  53 1.0 IS_GW_312:3128 (Squid/2.4.STABLE6)
```

Көрүнгөндөй 1751 суроо талап биздин прокси серверге чейин nt04 жана tasksmart прокси серверлерин кечип өткөн. Же болбосо 137 суроо талап биздин серверге reg3, cl, cache2 прокси серверлерден кечип өткөн.

Бул бөлүмдөгү маалымат эч жерге сакталбайт (оперативдик эсте болот), жана прокси сервер кайрадан жүктөлсө бул маалымат жок болот.

Forw_headers: X-forwarded-FOR request headers

Бул бөлүмдөгү маалымат SQUID прокси сервер ./configure – enable-forwd-via-db опциясы менен компиляцияланганда көрүнөт. X-forward-for баш аты

стандарттуу эмес, жалан гана SQUID прокси серверде колдонулат. Маалымат жогорку бөлүмдөгө окшош.

Shutdown: shutdown the SQUID process

Бул бөлүмдө прокси серверди узактан өчүрүү мүмкүнчүлүгү бар. Бул мүмкүнчүлүктүү уруксат кылуу үчүн squid.conf конфигурациондук файлында cahcemgr_passwd директивасы менен сыр сөздү аныктоо керек. Сыр сөз шифрленбейт, ошондуктан бул мүмкүнчүлүк өчүк болот.

Info: General run time information

Бул бөлүмдө прокси серведин иштөөсү боюнча абдан бай жана жалпы маалыматы берилет.

```
Squid Object Cache: Version 2.5.STABLE4

Start Time:     Mon, 22 Sep 2010 03:10:37 GMT

    Current Time:   Mon, 13 Oct 2010 10:25:16 GMT
    Connection information for squid:
        Number of clients accessing cache:      386
        Number of HTTP requests received:       12997469
        Number of ICP messages received:        16302149
        Number of ICP messages sent:    16310714
        Number of queued ICP replies:   0
        Request failure ratio:   0.00
        Average HTTP requests per minute since start:   423.7
        Average ICP messages per minute since start:    1063.2
        Select loop called: 400027445 times, 4.601 ms avg
```

Прокси серведин баштоо убактысы, учурдагы убакыт, баардык колдонуучулардын жалпы саны, баардык http-суроо талаптардын саны, баардык icp суроо талаптардын саны, минутада орточо суроо талап саны.

```
Cache information for squid:

        Request Hit Ratios:     5min: 22.6%, 60min: 25.8%

        Byte Hit Ratios:        5min: 24.6%, 60min: 38.7%

        Request Memory Hit Ratios:      5min: 0.7%, 60min: 1.4%

        Request Disk Hit Ratios:        5min: 6.0%, 60min: 12.4%

        Storage Swap size:      41457489 KB

        Storage Mem size:       10180 KB

        Mean Object Size:       14.43 KB

        Requests given to unlinkd:      0
```

Ошондой эле кэштин статистикасы, көлөмү, кэштэги маалыматтын колдонуучулардын суроо талабына туура келген обьектердин проценти ж.б.

```
Resource usage for squid:

        UP Time:        1840478.681 seconds

        CPU Time:       70571.874 seconds

        CPU Usage:      3.83%

        CPU Usage, 5 minute avg:        1.33%

        CPU Usage, 60 minute avg:       4.41%
```

```
        Process Data Segment Size via sbrk( ): 342739 KB

        Maximum Resident Size: 345612 KB

        Page faults with physical i/o: 65375
```

Прокси сервердин процессинин иштөө убактысы, колдонулган процессордук убакыттын көлөмү, процессорду колдонуу проценти, катуу дисктен маалымат окуу учурлары.

```
Memory usage for squid via mstats( ):

        Total space in arena:  415116 KB

        Total free:            129649 KB 31%
```

Процесс үчүн бөлүнгөн убактылуу эстин көлөмү, жана анын ичинен колдонулбай турган эстин көлөмү.

```
File descriptor usage for squid:

        Maximum number of file descriptors:   7372

        Largest file desc currently in use:    151

        Number of file desc currently in use:  105

        Files queued for open:                   0

        Available number of file descriptors: 7267

        Reserved number of file descriptors:   100
```

Store Disk files open: 0

Бир убакытта ачууга уруксаат берилген файл дескрипторлордун максимум саны, учурда колдонулуп жаткан файл дескриптордун саны, файл дескрипторлордун максимум колдонуу саны.

Filedesciptors: Process file descriptor allocation

Бөлүмдө учурда ачылып турган файл дескрипторлордун маалыматы көрсөтүлөт. Мисал таблица-9'ду көрсөтүлгөн:

Таблица-9 Учурда ачылып турган файл дескрипторлор

Id	Type	Toue	Nread	Nwrite	r.address	Desciption
3	File	0	0	0		/usr/local/logs/cache.log
6	File	0	0	3654789		/usr/local/logs/acces.log
13	Pipe	0	0	2345987		Unlinkd->squid
23	Socket	24	3545785	5658	62.35.1.22	http://mp3.com

TYPE-чоңдугу декскриптордун түрүн көрсөтөт (file, pipe, socke). Файлдар кэширлөөдө, лог жазылып жатканда ачылат, pipe- болсо эки процесс бир бирине маалымат алмашуу үчүн ачылат, socket – бул дагы процесстердин маалымат алмашуу жолу, бирок HTTP же FTP байланышта клиент-сервер арасында.

TOUT-чоңдугу дескриптордун колдонулбаган убакытын көрсөтөт. Файл жана pipe типтери үчүн дайыма 0-го барабар болот. Ал эми socket'тер үчүн бул чоңдук максималдуу мааниге жеткенде жабылуу функция чакырылат.

NREAD-чоңдугу дескриптордон N - байт окулгандыгын көрсөтөт.

NWRITE-чоңдугу дескриптерге N - байт жазылгандыгын көрсөтөт.

REMOTEADDRESS-сокеттин ачылган узактагы адресин көрсөтүлөт.

Objects: All cache objects

Бул бөлүмдө прокси серведин кэшинде сакталган баардык обьектердин тизмесин көрүүгө болот. Кэширленген обьектердин саны абдан көп болгондуктан маалыматты көрсөтүүдө бир нече убакыт алышы мүмкүн.

Vm_objects: In-memory and In-transit objects

Бул бөлүм жогорку «All cache objects» бөлүмгө окшош, бирок учурда оперативдик эсте сакталып жаткан же болбосо учурда талапта суралып жаткан обьетилердин тизмесин камтыйт.

```
KEY 5107D49BA7F9C6BA9559E006D6DDC4B2

        GET
http://www.rpgplanet.com/ac2hq/cartography/dynamic/LinvakMassif.jpg

        STORE_PENDING NOT_IN_MEMORY SWAPOUT_WRITING PING_DONE

        CACHABLE,DISPATCHED,VALIDATED

        LV:1043286120 LU:1043286122 LM:1036015230 EX:-1

        4 locks, 1 clients, 1 refs

        Swap Dir 1, File 00X31BD9

        inmem_lo: 184784

        inmem_hi: 229840

        swapout: 229376 bytes queued

        swapout: 229509 bytes written
```

io: Server –side network read() size histograms

Бул бөлүмдө сервер тараптагы 4-протоколдордун HTTP, FTP, Gopher, WAIS гистограммасын камтыйт

Counters: Traffic and resource counters

SQUID прокси сервер атайын ички счетчиктерди тейлейт. Бул счетчиктер массив түрүндө сакталат жана жөнөкой адам түшүнө албай турган түрдө сакталат, бул бөлүмдү атайн компьютердик программа менен окутуу дагы ыңгайлуу болот.

Peer_select: Peer selection algorithms

Digest_stat: Cache digest and icp blob

Бул бөлүмдөгү маалымат жалаң гана SQUID прокси серверин иштетүүчү программисттерге кызык болот. Жана прокси сервердин ички статистикасы көрсөтүлөт.

5min: 5 minute avarage of counters

60min: 60 minute avarage of counters

Бул бөлүмдөгү маалымат жалаң гана SQUID прокси серверин иштетүүчү программисттерге кызык болот. Жана прокси сервердин ички статистикасы көрсөтүлөт.

Active_requests: Client-side active requests

Бул бөлүмдөгү маалымат колдонуучулардын айпи адреси, учурдагы суроо-талаптары, жана байланыштын адреси, байланыштын убактысы, веб трафиктин көлөмү. Биздин учурдагы изилдөөдө абдан бай жана толук маалымат бере алатураган бөлүм болуп саналат. Ошондуктан «active_requests» бөлүмүн тагыраак жана терең карап кетсек:

```
Connection: 0x84ecd10

        FD 132, read 1273, wrote 12182

        FD desc: http://www.squid-cache.org/Doc/FAQ/FAQ.html

        in: buf 0xa063000, offset 0, size 4096

        peer: 206.168.0.9:1058
```

```
        me: 192.43.244.42:3128

        nrequests: 3

        defer: n 0, until 0

uri http://www.squid-cache.org/Doc/FAQ/FAQ.html

log_type TCP_MISS

out.offset 0, out.size 0

req_sz 392

entry 0x960c680/3B49762ABF444D80B6465552F6CFAD4C

old_entry 0x0/N/A

start 1066036250.669955 (2.240814 seconds ago)
```

Connection – чоңдугу, прокси сервердин ички эсинде сакталган учурдагы байланыштын структурасына көрсөтөт.

FD – чоңдугу, учурдагы байланыштын TCP файл дескрипторун жана бул дескриптордон окулган/жазылган маалыматтын көлөмүн камтыйт. (байт менен)

FD desc – байланыштын кыскача түшүнүгү, дайыма uri адрести камтыйт

In – ички буффердин адреси, жана анын көлөмү. Бул жерге сокеттен келген маалымат жазылат.

Peer – учурда байланышып жаткан хостун сокет адреси.

Me – учурдагы байланыштын ички сокет адреси.

Nrequests – учурдагы байланыштын негизинде келген ийгиликтүү жооптордун саны, бирден айырмаланган сан болсо демек учурдагы байланыш туруктуу/үзгүлтүксүз/persistent болгондугун билдирет.

Differ – учурдагы сокеттен прокси сервер тараптан маалымат окулбай калган убакыт өлчөмүн көрсөтөт.

Uri – колдонуучунун сурап жаткан маалыматынын толук URI адреси.

Log_type – прокси сервердин учурдагы суроо-талаптын абал кодун билдирет.

Out_offset – колдонуучуга жибериле турган маалыматтын кемтиги.

Req_sz – колдонуучунун учурдагы HTTP суроо-талабынын көлөмү. Эгерде байланыш туруктуу болсо анда баардык HTTP суроо-талабтын көлөм эмес учурдагы көлөмдү гана көрсөтөт.

Entry – ийгиликтүү жооптун струтурасынын эстеги адреси.

Old_entry – кэште эсте сакталган ийгиликтүү жооптун структурасынын адреси (жаны жооп менен салыштыруу үчүн колдонулат)

Start – учурдагы суроо талаптын баштоо убактысы.

Store_digest: Store Digest

Бул бөлүмдөгү маалымат SQUID прокси сервер ./configure – enable-cache-digest опциясы менен компиляцияланганда көрүнөт. Прокси сервердин ички кэш каталогунун статистикасын камтыйт:

```
store digest: size: 620307 bytes

        entries: count: 324806 capacity: 992490 util: 33%

        deletion attempts: 0

        bits: per entry: 5 on: 1141065 capacity: 4962456 util: 23%

        bit-seq: count: 1757902 avg.len: 2.82

        added: 324806 rejected: 611203 ( 65.30 %) del-ed: 0

        collisions: on add: 0.08 % on rej: 0.07 %
```

Storedir: Store Directory Stats

Бөлүмдө прокси сервердин кэшти сактоо системасынын статистикасы көрсөтүлөт. Эң башта жалпы кэш системасынын маалыматы келтирилет.

```
Store Directory Statistics:

Store Entries          : 2873564

Maximum Swap Size      : 46080000 KB

Current Store Swap Size: 41461672 KB

Current Capacity       : 90% used, 10% free
```

Store entries – «store entry» обьектилердин саны, көбүнчөлөрү катуу дискте сакталат.

Maximum swap size – Баардык cache_dir папкалардын көлөмү

Current Capacity – кэш үчүн бөлүнгөн катуу дисктеги бош орундун проценттик көрсөткүчү.

Жалпы маалыматтан кийин ар бир cache_dir папкалары үчүн толук статистика келтирилет:

```
Store Directory #1 (diskd): /cache1

FS Block Size 1024 Bytes

First level subdirectories: 16

Second level subdirectories: 64

Maximum Size: 15360000 KB

Current Size: 13823996 KB

Percent Used: 90.00%

Filemap bits in use: 958439 of 2097152 (46%)

Filesystem Space in use: 14030485/17370434 KB (81%)
```

```
Filesystem Inodes in use: 959440/4340990 (22%)

Flags: SELECTED

Pending operations: 0

Removal policy: lru
```

LRU reference age: 23.63 days

Store_check_cachable_stats: storeCheckCachable() stats

storeCheckCachable функциясынын статистикасын камтыйт. Бул функция ар дайым ийгиликтүү жооп келгейн сайын чакырылат. Жана келген жоопту кэширлөө же кеширлебөө чечими чыгарылат.

Store_io: Store IO interface stats

Бөлүмдө жаңы обьекттин кэште сактоо үчүн дискте орун айыруу статистикасы

```
Store IO Interface Stats

create.calls 2825670

create.select_fail 0

create.create_fail 0

create.success 2825670
```

create calls – катуу дискте жаңы файл жаратуучу функциянын жакыруу саны

create.select_fail – жаңы файл жаратууда каталуу аяктоо операциялардын саны. Эгерде баардык кэш директориялар учурда бош болбосо.

Create.create_fail - жаңы файл жаратууда каталуу аяктоо операциялардын саны. Катуу диск демондун жаны файл ачууга каталуу жооп менен аякташы.

Create.success – жаңы файл жаратууда ийгиликтүү аяктоо операциялардын саны.

Pconn: Persistent connetion utilization histograms

Туруктуу/үзгүлтүксүз байланыштардын гистограммасын камтыйт

Refresh: refresh algorithm statistics

Кэште сакталып жаткан обьектилердин жаңылыгынын статистикасы. Прокси сервер кэштеги обьектлердин «жашын» текшерип турат. Жана эскирген обьекттер өчүрүлөт же жаңдан суралат.

Forward: request forwarding statistics

Бөлүмдө прокси сервери тараптан «жиберүү-forwarding» операциясынын статистикасы көрсөтүлөт.

Client_list: Cache client list

Бөлүмдө ар бир колдонуучунун IP адресине жараша жалпы маалыматы көрсөтүлөт: Баардык http - суроо талаптардын саны, алардын ичинен кэштеги маалыматтын дал келүүсүнүн проценти, келбөөсүнүн проценти ж.б

```
Address: 206.168.0.9
Name: 206.168.0.9
Currently established connections: 0

    ICP Requests 59000
        UDP_HIT                1609   3%
        UDP_MISS              57388  97%
        UDP_INVALID               3   0%
    HTTP Requests 11281
        TCP_HIT                 656   6%
        TCP_MISS               3464  31%
        TCP_REFRESH_HIT        4477  40%
        TCP_REFRESH_MISS        767   7%
        TCP_CLIENT_REFRESH_M    397   4%
        TCP_IMS_HIT            1082  10%
        TCP_SWAPFAIL_MISS         7   0%
        TCP_NEGATIVE_HIT         13   0%
        TCP_MEM_HIT             418   4%
```

Netdb: Network measurement database

Бул бөлүмдөгү маалымат SQUID прокси сервер ./configure – enable-icmp опциясы менен компиляцияланганда көрүнөт. ICMP пакеттердин санын көрүүгө болот.

Server_list: Peer cache statistics

Прокси сервердин башка «кошуна» прокси серверлердин статистикасы көрсөтүлөт:

```
Sibling    : pa.us.ircache.net/3128/4827
Flags      : htcp
Address[0] : 192.6.19.203
Status     : Up
AVG RTT    : 14 msec
OPEN CONNS : 19
LAST QUERY : 4 seconds ago
LAST REPLY : 4 seconds ago
PINGS SENT : 9119
PINGS ACKED: 9115 100%
FETCHES    : 109   1%
IGNORED    : 9114 100%
Histogram of PINGS ACKED:
        Misses      9114 100%
        Hits           1   0%

keep-alive ratio: 100%
```

Type – кошуна прокси сервердин тибин аныктайт (parent, sibling, multicast). Жана хостун аты, http порту, htcp порту көрсөтүлөт.

Address – кошуна прокси сервердин IP адреси

Status – кошуна сервердин статусу

Avg RTT – кошуна прокси серверге чейин тармак боюнча жиберилген маалыматтын орточо короткон убакыт көлөмү.

Open conns – кошуна прокси сервер менен учурдагы ачык байланыштардын саны

Last query – акыркы icp/htcp суроо талаптын жиберүү убактысы

Last replay – акыркы icp/htcp жооптун келүү убактысы

Pings sent – icp/htcp суроо талаптардын саны

Pings acked – icp/htcp суроо талаптардын ийгиликтүү кайта келүүсү

None_peers: List of Unknown sites sending ICP messages

Бул бөлүмдө белгисиз жана чоочун колдонучулардан келген ICP пакет жана IP адрестери көрсөтүлөт.

2.5 Cache Manager'ге чектөөлү кириш

Жогоруда көрүнгөндөй Cache manager абдан маанилүү жана пайдалуу маалыматтарды камтыйт, кээ бир учурда бөлүмдөрдө критикалык маалымат берилет. Бул маалыматты жалаң гана администратор көрө алыш керек жана сырткы колдонуучуларга кирүү-окуу мүмкүнчүлүктүн жабылуусу зарыл. "Cache client list" бөлүмүндө баардык колдонуучулардын IP адрестери камтылат, "Process filedescriptor" бөлүмүндө учурдагы суроо-талаптын адреси ал эми "Current Squid configuration" бөлүмүндө болсо критикалык маалымат – конфигурациялык директивалар, сыр сөздөр, уруксаат эрежелери камтылат.

Cache manager'ди сырткы кол салуудан сактоо үчүн бул бөлүмгө жалаң гана ишеништүү IP адрестердин тизмесин түзүп төмөнкүдөй саптар конфигурациялык файлга жазылыш керек:

```
acl Manager proto cache_object
acl Localhost src 127.0.0.1/255.255.255.255
http_access allow Manager Localhost
http_access deny Manager
```

Мисалда жалан гана локалхосттон (localhost) Cache manager'ге кирүү мүмкүн (127.0.0.1) ал эми калгандарына болсо кирүү жабык болот.

Ошондой эле Cache manager'дин ар бир бөлүмдөрүнө сыр сөз аныктоо мүмкүнчүлүгү бар. Анын үчүн конфигурациалык файлга (squid.conf) төмөнкүдөй сап жазуу зарыл:

```
Cachemgr_passwd PASSWORD filedescriptors client_list netdb
```

Ал эми кээбир бөлүмдөрдү такыр эле өчүрүп салуу зарыл болсо:

```
Cachemgr_passwd disable netdb
```

Сыр сөздү алып салуу:

```
Cachemgr_passwd none offline_toggle
```

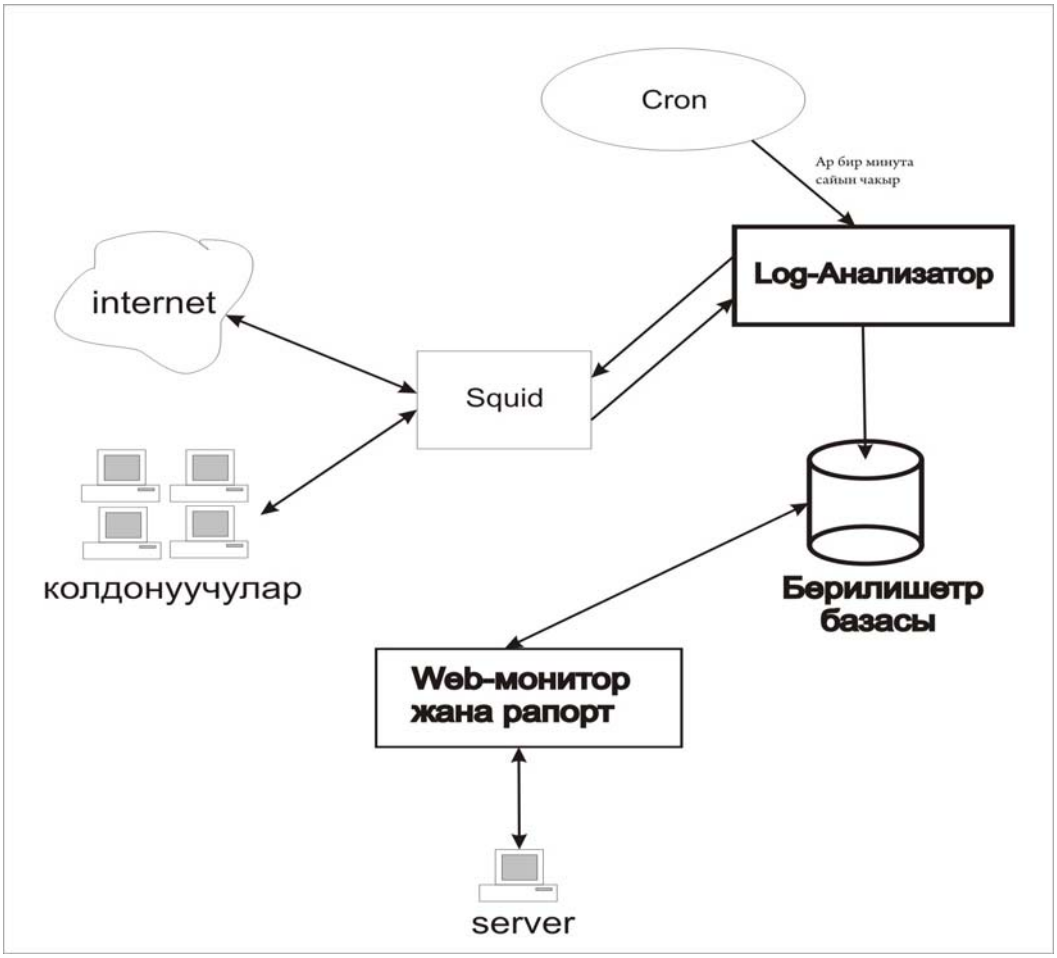
Баардык бөлүмдөргө бирдей сыр сөз берүү үчүн:

```
Cachemgr_passwd PASSWORD all
```

БӨЛҮМ – 3 ПРОТОТИПТӨӨ, СЫНОО

Жогоруда каралган лог-анализаторлордун кемчиликтерин эске алып, дагы оптималдуу лог файлдарды иштетүү жолу менен төмөнкү сүрөт-3.1дей системага ээ боло алабыз. Бул жерде лог анализатор ар бир минута сайын КРОН утилитасы жардамы менен чакырылып турат, жана мындайча лог файлдын өтө бат өсүп кетишине келтирбейт. Мындайча лог анализатордун берген маалыматы эң жаңы болуп турат. Ошондой эле баардык маалымат БДга сакталат, жана веб интерфейс аркылуу бул маалыматты көрүүгө болот.

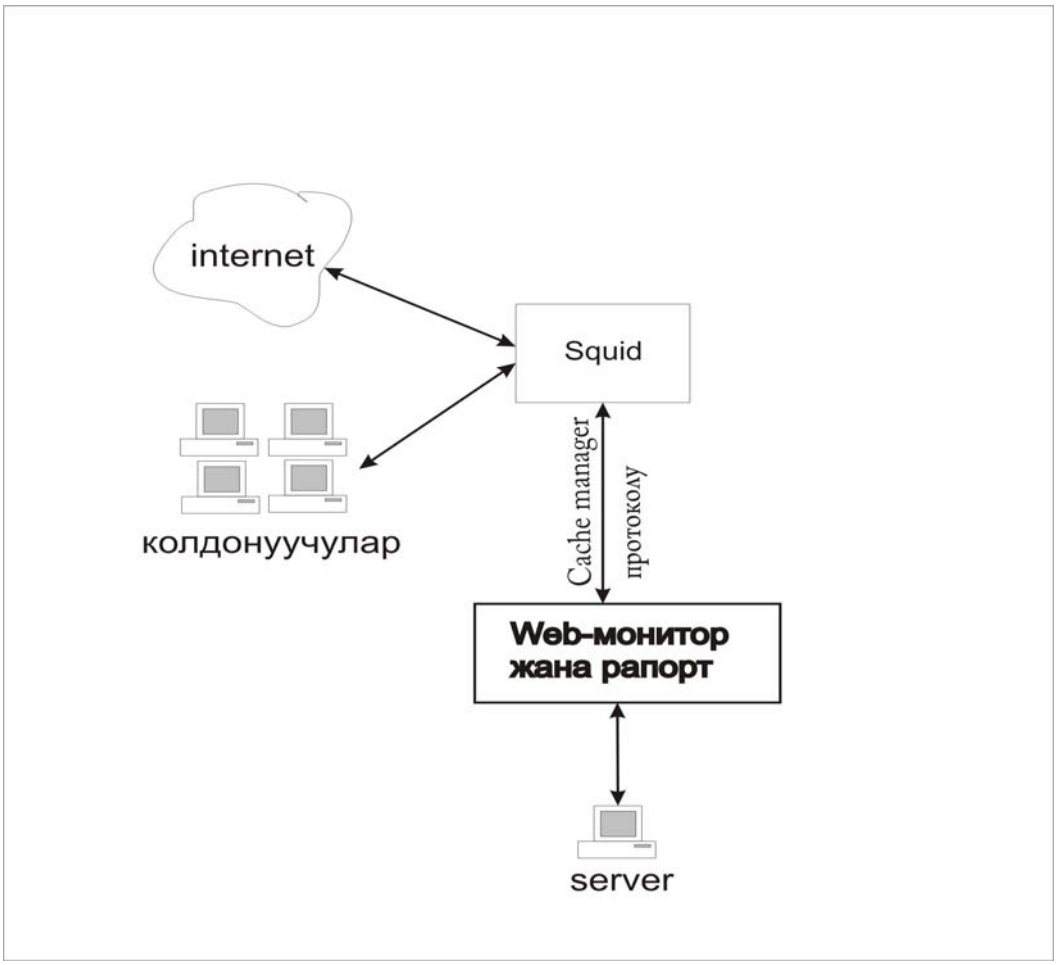
Сүрөт – 3.1 (Лог-анализатордун негизиндеги система)



Ошондой эле жогоруда каралган cache_manager протоколун эске алып, протоколдун мүмкүнчүлүгүн колдонуп биздин алдыбызга коюлган максатка

башка жолу менен жетишүү мүмкүн экендигин көрө алабыз. Жана төмөнкү сүрөт-3.2дей архитектурага ээ болобуз.

Сүрөт – 3.2 Cache manager негизиндеги система



Сүрөт -3.1 де көрсөтүлгөн системада заматта мониторинг бир аз жасалма (жана көп ресурсту колдонуу) жолу менен жетишет. Бирок маалымат БД сында сакталган үчүн абдан ыңгайлуу система болуп саналат. Ал эми 3.2-чи сүрөттө болсо тескеринче заматта мониторинг «табигий» жол менен жана аз ресурс менен жетишет. Бирок маалыматты сактоо мүмкүнчүлүгү жокко эсе.

Албетте эң оптималдуу жол менен заматта веб трафикти аңдоо жана ошондой эле баардык статистикалык маалыматтын сактоо мүмкүнчүлүгүн берген система жаратуу зарыл экендиги көрүнүп жатат. Бул максатка жетишүү үчүн жогоруда сүрот-3.1 жана сүрөт-3.2 де көрсөтүлгөн системалардын симбиозун жаратуу зарыл. Мындай системаларды ар түрдүү жол менен жана ар

кандай методтор менен жаратуу аракеттери бүгүнкү күндө байкалууда. Мисал катары Antikor системасын карасак болот. Antikor – толугу менен веб-интерфейстин жардамы менен башкарылат, жана онлайн мониторинг менен веб трафик статистикасын жүргүзөт. Бирок бул системанын кемчилиги акчага сатылат, жана ошондой эле өзүнүн компьютердик жабдуулары менен бирге келет. Дагы бир нече «аты» чыга элек системалар да бар, алар көптөгөн каталарды камтыйт жана алардын абалы «иштетүү-разработка» стадиясында.

Бүгүнкү күнгө чейин бир нече операциондук системаларда SQUID прокси сервери орноштурулду жана тестирленди. Булар FreeBSD 8.0, Fedora Core 10, Fedora Core 11 жана Ubuntu 7.0, Debian 5.0 операциондук системалар. Айрыкча FreeBSD 8.0 операциондук системасында токтолуп кетебиз.

Аягына чыккан кадамдар:

- Учурда FreeBSD 8.0 операциондук система колдонууда.
- SQUID конфигурацияланып ички тармактагы колдонуучуларга Интернет сервисин берип жатууда.
- SQSTAT жана MYSAR системалары орнотулуп, тестирленип ийгиликтүү иштеп жатышат.
- Mkfifo жардамы менен named pipe түзүлдү, жана стандарттык /var/log/squid/access.log файлынын ордуна /var/log/squid/access.log атту named pipe жаратылды. Эч нерсени байкабаган SQUID прокси сервери бур named pipe га баардык статистикалык маалыматтарды катасыз жазууда.

3.1 Cache manager негизинде заматта мониторинг прототиби

Сүрөт – 3.1 көрсөтүлгөн системанын прототибин FreeBSD операциондук системасына жазылган SQSTAT деген модульдун базасында аткарууну чечтик. SQSTAT модулу 28/04/2006-жылы украиналык программист тарабынан жазылып, sourceforce.net сайтына ачык коду менен илинген жана кыска убакыттын ичинде популярдуу болгон. FreeBSD 6.0 версиясындан баштап бул модуль операциондук системанын порт коллекциясы менен кошо келет. 07/10/2006 убактысынан баштап бул модулдун иштетилүүсү токтотулат.

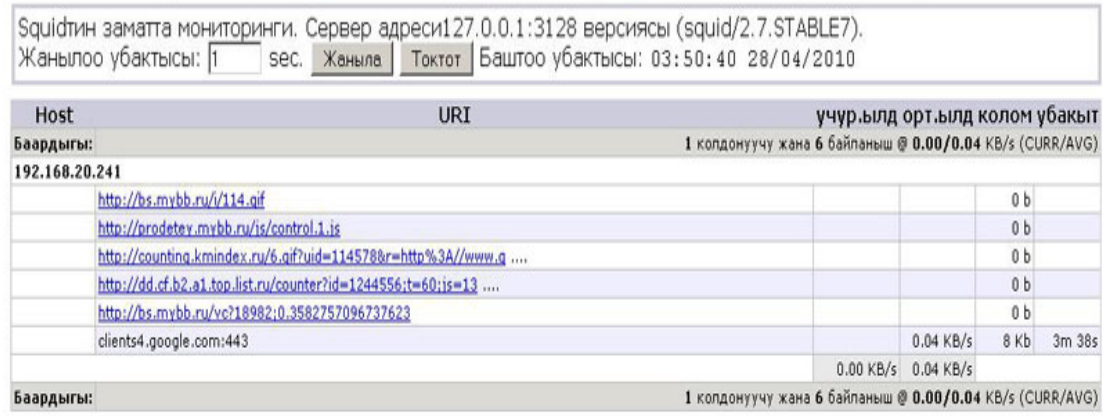
Төмөнкү сүрөт-3.3тө SQSTAT классынын UML диаграммасы көрсөтүлгөн. Бул класс PHP 4.0 версиясында жазылган. Ошондуктан сүрөттө көрүнгөндөй класстын ички өзгөрмөлөрү «ачык-public» жарыялган. (+ символу менен башталат)

Сүрөт – 3.3 sqstat классынын UML диаграммасы

SQSTAT тын өзгөртүлгөн толук коду Тиркеме-2 де берилген. Өзгөртүүлөрдүн негизин модулга жаңы мүмкүнчүлүк киргизүү жана кыргызча тилге которуу максатын түзөт.

Модул биздин системага (FreeBSD 8.0 RELEASE, Squid Cache 2.7 STABLE7, MySQL 5.0.86, Apache 2.2.13, GCC 4.2.1, PHP 5.2.11) орнотулуп ийгиликтүү тестерди өттү.

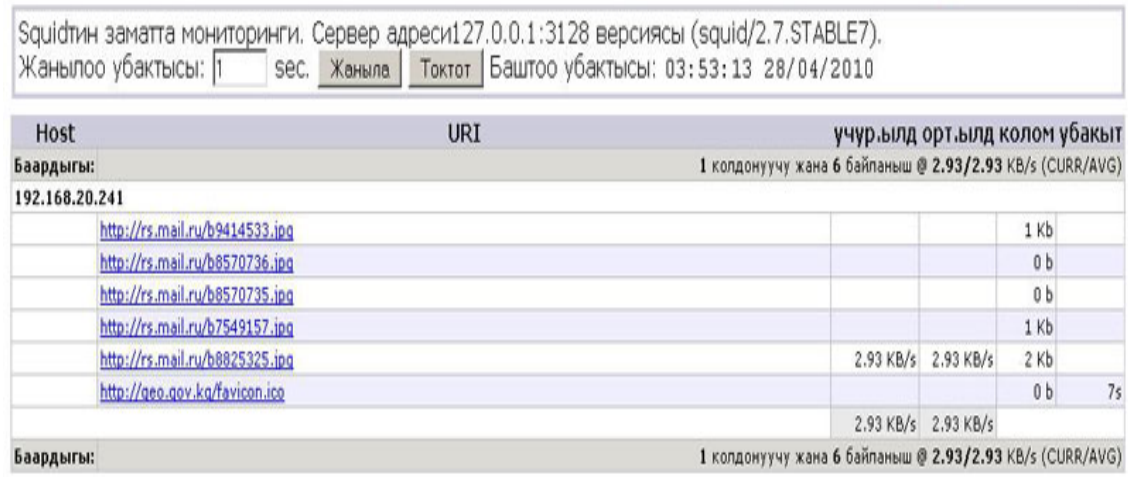
Сүрөт 3.4 (Cache manager дин негизинде иштеген анализатордун скриншоту)



Squidтин заматта мониторинги. Сервер адреси127.0.0.1:3128 версиясы (squid/2.7.STABLE7).
Жанылоо убактысы: 1 sec. Жаңыла Токтот Баштоо убактысы: 03:50:40 28/04/2010

Host	URI	үчүр.ылд	орт.ылд	колом	убакыт
Баардыгы:	1 колдонуучу жана 6 байланыш @ 0.00/0.04 KB/s (CURR/AVG)				
192.168.20.241					
	http://bs.mybb.ru/i/114.gif			0 b	
	http://prodetev.mybb.ru/js/control.1.js			0 b	
	http://counting.kmindex.ru/6.gif?uid=114578&r=http%3A//www.g			0 b	
	http://dd.cf.b2.a1.top.list.ru/counter?id=1244556;t=60;js=13			0 b	
	http://bs.mybb.ru/vc?18982;0.3582757096737623			0 b	
	clients4.google.com:443		0.04 KB/s	8 Kb	3m 38s
		0.00 KB/s	0.04 KB/s		
Баардыгы:	1 колдонуучу жана 6 байланыш @ 0.00/0.04 KB/s (CURR/AVG)				

Сүрөт 3.5 (Cache manager дин негизинде иштеген анализатордун скриншоту)

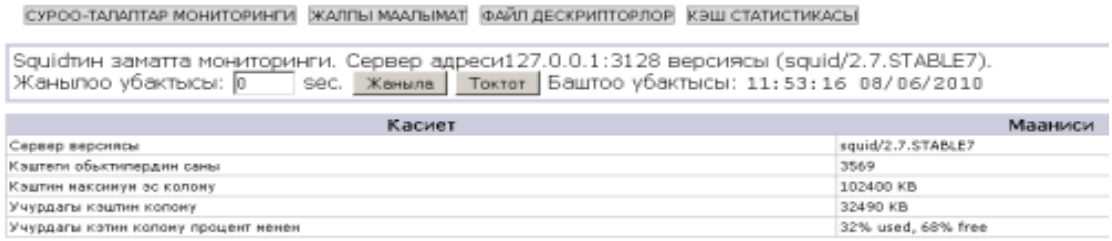


Squidтин заматта мониторинги. Сервер адреси127.0.0.1:3128 версиясы (squid/2.7.STABLE7).
Жанылоо убактысы: 1 sec. Жаңыла Токтот Баштоо убактысы: 03:53:13 28/04/2010

Host	URI	үчүр.ылд	орт.ылд	колом	убакыт
Баардыгы:	1 колдонуучу жана 6 байланыш @ 2.93/2.93 KB/s (CURR/AVG)				
192.168.20.241					
	http://rs.mail.ru/b9414533.jpg			1 Kb	
	http://rs.mail.ru/b8570736.jpg			0 b	
	http://rs.mail.ru/b8570735.jpg			0 b	
	http://rs.mail.ru/b7549157.jpg			1 Kb	
	http://rs.mail.ru/b8825325.jpg	2.93 KB/s	2.93 KB/s	2 Kb	
	http://geo.gov.kg/favicon.ico			0 b	7s
		2.93 KB/s	2.93 KB/s		
Баардыгы:	1 колдонуучу жана 6 байланыш @ 2.93/2.93 KB/s (CURR/AVG)				

Көрүнгөндөй, учурдагы активдүү колдонуучулардын тизмесин жана алардын окуп-көрүп жаткан рессурстардын адрестерин көрүүгө болот. Ар бир ресурстун көлөмүн, жүктөлүү ылдамдыгын жана убактысын көрсөтөт. Ошондой эле сервердин ичики компоненттеринин мониторинг мүмкүнчүлүгү кошулган, жана ички процесстердин агуусуна байкоо жүргүзүү мүмкүн (сүрөт 3.6, 3.7).

Сүрөт 3.6 (MySAR лог анализаторунун негизги бет скриншоту)

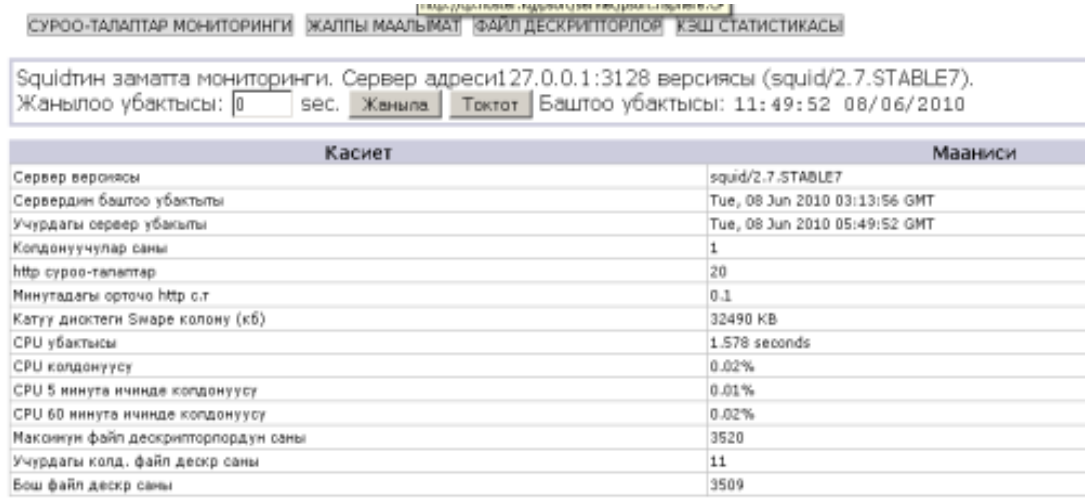

СУРОО-ТАЛАПТАР МОНИТОРИНГИ ЖАЛПЫ МААЛЫМАТ ФАЙЛ ДЕСКРИПТОРЛОР КЭШ СТАТИСТИКАСЫ

Squidтин заматта мониторинги. Сервер адреси127.0.0.1:3128 версиясы (squid/2.7.STABLE7).
Жанылоо убактысы: 0 sec. Жаныла Токтот Баштоо убактысы: 11:53:16 08/06/2010

Касиет	Мааниси
Сервер версиясы	squid/2.7.STABLE7
Кэштеги обьектилердин саны	3569
Кэштин максимун эс колому	102400 KB
Учурдагы кэштин колому	32490 KB
Учурдагы кэтин колому процент менен	32% used, 68% free

Баардык бул маалымат Тиркеме-1 деги массивдин негизинен келип чыгат, ал эми Тиркеме-1 деги массив cache manager протоколунун жардамы менен прокси серверден суроо талап аркылуу алынган.

Сүрөт 3.7


СУРОО-ТАЛАПТАР МОНИТОРИНГИ ЖАЛПЫ МААЛЫМАТ ФАЙЛ ДЕСКРИПТОРЛОР КЭШ СТАТИСТИКАСЫ

Squidтин заматта мониторинги. Сервер адреси127.0.0.1:3128 версиясы (squid/2.7.STABLE7).
Жанылоо убактысы: 0 sec. Жаныла Токтот Баштоо убактысы: 11:49:52 08/06/2010

Касиет	Мааниси
Сервер версиясы	squid/2.7.STABLE7
Сервердин баштоо убактысы	Tue, 08 Jun 2010 03:13:56 GMT
Учурдагы сервер убактысы	Tue, 08 Jun 2010 05:49:52 GMT
Колдонуучулар саны	1
http суроо-талаптар	20
Минутадагы орточо http с.т	0.1
Катуу дисктеги Swape колону (кб)	32490 KB
CPU убактысы	1.578 seconds
CPU колдонуусу	0.02%
CPU 5 минута ичинде колдонуусу	0.01%
CPU 60 минута ичинде колдонуусу	0.02%
Максимун файл дескрипторлордун саны	3520
Учурдагы колд. файл дескр саны	11
Бош файл дескр саны	3509

3.2 Лог-анализатор негизинде заматта мониторинг прототиби

SQUID прокси серведин access.log файлын иштетүүчү лог-анализатор негизиндеги прототипти MYSAR системасынын негизинде аткарууну чечтик. Себеби бул система PHP тилинде жазылган, жана MYSQL берилиштер базасын колдонот. Ал эми бул технологиялар бекер жана абдан кеңири таралганы белгилүү.

- Бекер. General Public Licence лицензиясы алдында таркатылгандыгы үчүн Maysar баардык колдонуучуларга бекер.
- Динамикалык. Эч кандай файлдардын тизмесин жаратпастан отчет жаратуу мүмкүнчүлүгүн камтыйт, жана түздөн түз БДсынан генерациялайт.

- Дээрлик бат. Отчетторду көрүү үчүн сааттар күтүү кереги жок, отчеттордун генерациясы ылдам, жана каалаган убакытта кол алдында.
- Ташымалдуу. Таза PHP тилинде жазылгандыгы үчүн, жана кошумча китепканаларды колдонбогон үчүн дээрлик баардык Unix үй бүлөсүндөгү операциондук системаларда иштей алат. Жана эч кандай модификацияны талап кылбайт.
- Ийилгичтүү. Коду ачык болгондуктан каалаган колдонуучу каалаган өзгөртүү киргизе алат, жана сырткы көрүнүштү, отчет жаратуу процессин өрзгөртүү мүмкүн.
- Стабилдүү. PHP, MySQL кодтору убакыт боюнча текшерилип стабилдүү калыбка келген.
- Жеңил. Абдан оңой жүктөлөт, жана ичине курулган жүктөөчү менен бирге келет.

3.3 Берилиштер базасынын структурасы

Алты таблица түзүлдү, алардын кыскача аныктамалары төмөндөгүчө:

config: Бул таблицада системанын настройкалары сакталат. Сортоо түрү, акыркы тазалоо убактысы, таймстэмп, лог файлдын жолу, жана башкалар.

hostnames: Баардык колдонуучулардын берилиштери (клиенттердин) бул таблицада сакталат. IP адресс, компьютердин ДНС жазуугадагы мааниси жана кыскача түшүндүрмөлөр.

sites: Колдонуучулар ачып көргөн сайттардын тизмеси, тарыхы.

traffic: Интернет трафик, SQUID тин жараткан лог файлына жараша. Башкача айтканда raw-trafic.

trafficSummaries: Бул таблицада бизге маанилүү болгон берилиштер сакталат, баардык рапортко керектүү берилиштердин саналуу, индекстелген тобу. Рапорт жаратууда индекстелген таблица колдонуу абдан ыңгай болот

users: Колодонучулар таблицасы.

MySAR БД-ныи MySQL коду

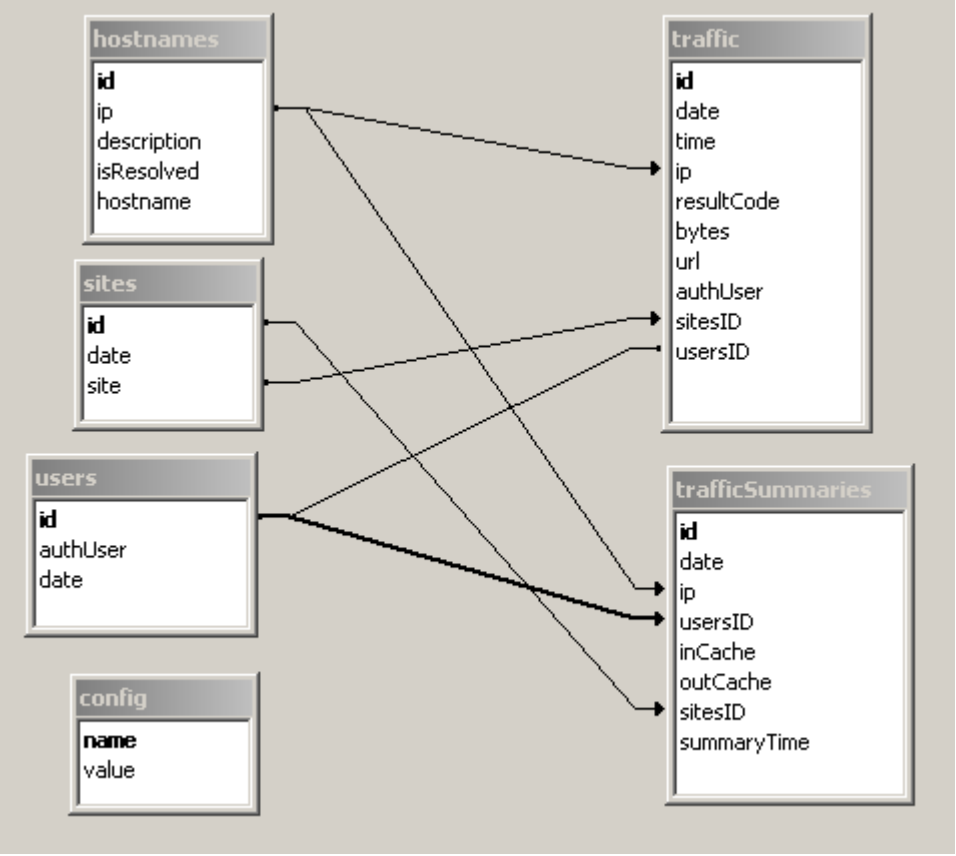
```
CREATE TABLE IF NOT EXISTS config (
  name varchar(255) NOT NULL default '',
  `value` varchar(255) NOT NULL default '',
  UNIQUE KEY name (name)
);
CREATE TABLE IF NOT EXISTS hostnames (
  id bigint(20) unsigned NOT NULL auto_increment,
  ip int(10) unsigned NOT NULL default '0',
  description varchar(50) NOT NULL default '',
  isResolved tinyint(3) unsigned NOT NULL default '0',
  hostname varchar(255) NOT NULL default '',
  PRIMARY KEY  (id),
  KEY isResolved (isResolved),
  KEY ip (ip)
);
CREATE TABLE IF NOT EXISTS trafficSummaries (
  id bigint(20) unsigned NOT NULL auto_increment,
  `date` date NOT NULL default '0000-00-00',
  ip int(10) unsigned NOT NULL default '0',
  usersID bigint(20) unsigned NOT NULL default '0',
  inCache bigint(20) unsigned NOT NULL default '0',
  outCache bigint(20) unsigned NOT NULL default '0',
  sitesID bigint(20) unsigned NOT NULL default '0',
  summaryTime tinyint(3) unsigned NOT NULL default '0',
  PRIMARY KEY  (id),
  UNIQUE KEY date_ip_usersID_sitesID_summaryTime
(`date`,ip,usersID,sitesID,summaryTime)
);
```

```
CREATE TABLE IF NOT EXISTS traffic (
  id bigint(20) unsigned NOT NULL auto_increment,
  `date` date NOT NULL default '0000-00-00',
  `time` time NOT NULL default '00:00:00',
  ip int(10) unsigned NOT NULL default '0',
  resultCode varchar(50) NOT NULL default '',
  bytes bigint(20) unsigned NOT NULL default '0',
  url text NOT NULL default '',
  authuser varchar(30) NOT NULL default '',
  sitesID bigint(20) unsigned NOT NULL default '0',
  usersID bigint(20) unsigned NOT NULL default '0',
  PRIMARY KEY  (id),
  KEY date_ip_sitesID_usersID (`date`,ip,sitesID,usersID)
);

CREATE TABLE IF NOT EXISTS users (
  id bigint(20) unsigned NOT NULL auto_increment,
  authuser varchar(50) NOT NULL default '',
  `date` date NOT NULL default '0000-00-00',
  PRIMARY KEY  (id),
  UNIQUE KEY date_authuser (`date`,authuser),
  KEY authuser (authuser)
);
CREATE TABLE IF NOT EXISTS sites (
  id bigint(20) unsigned NOT NULL auto_increment,
  `date` date NOT NULL default '0000-00-00',
  site varchar(255) NOT NULL default '',
  PRIMARY KEY  (id),
  UNIQUE KEY date_site (`date`,site)
);
```

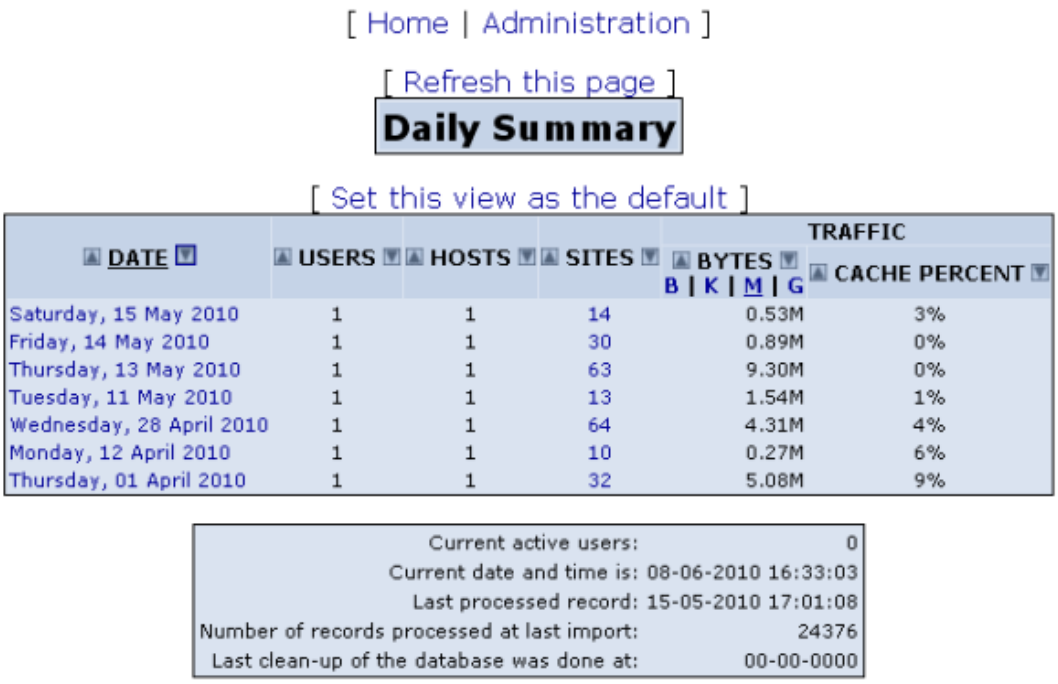
БД таблицалардын байланыш катышы **т**өмөнкү сүрөттөгүдөй

Сүрөт 3.8



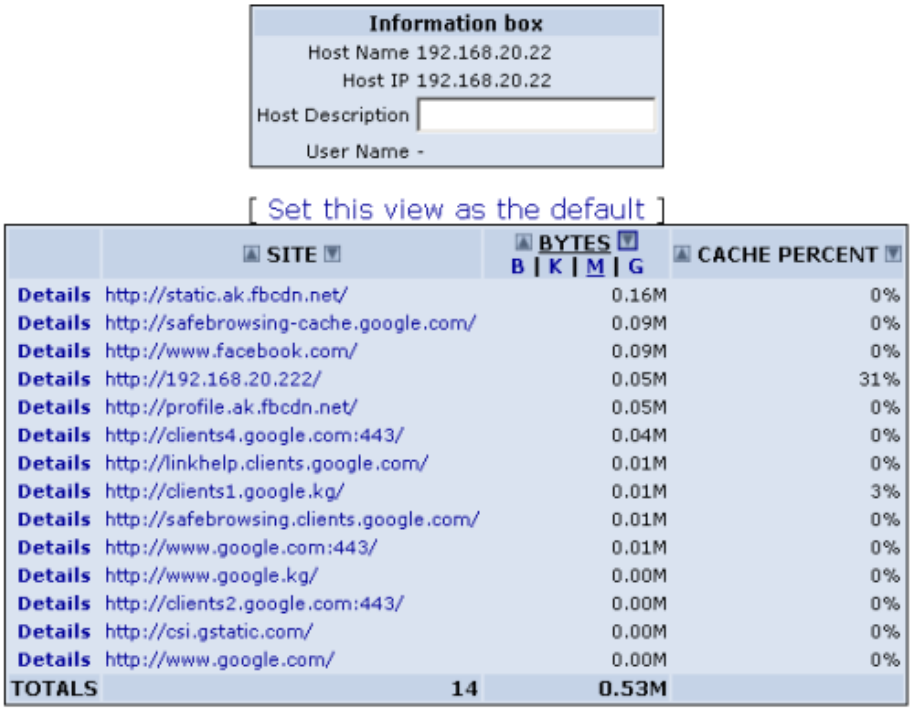
MaySAR абдан толук жана бай анализ жүргөзө алат, жана төмөнкү сүрөттөрдө көрүнгөндөй отчеттордун бир нече түрүн камтыйт. Дээрлик баардык анализаторлордун кодун изилдегенге аракет жазалды, жана MySAR инструменти ыңгай жана оңой өзгөртүүгө мүмкүнчүлүгүн камтыган аназаторлордон болуп чыкты. Учурдагы изилдөөдө эталон катары алынып, ыңгайлуу прототип функциясын аткарды.

Сүрөт 3.9 (MySAR лог анализаторунун негизги бет скриншоту)



Отчеттордун ичинде колдонуучунун компьютеринин аты боюнча фильтрлөө мүмкүнчүлүгү бар.

Сүрөт 3.10 (: MySAR лог анализаторунун башкы бет скриншоту)



Жыйынтык

SQUID прокси сервери бүгүнкү күндө эң популярдуу кэш сервер болуп саналат. Жана бул сервер келечекте дагы көптөгөн мекемелер тарабынан колдоноору шексиз. Эң бат, жана көптөгөн операциондук системаларада иштей ала турган сервер. SQUID тармактагы Интернет маалымат жүктү азайтууда чоң рол ойнойт. Веб баракчардын бат жүктөлүүсүнө жана керек болсо веб сервердин иштөөсүнө жана веб сервердин жүгүн азайтууда чоң көмөк көрсөтөт. Ошондой эле ички колдонуучуларды сырткы салдыруулардан коргоодо жана коопсуздукту сактоодо эң эффективдүү инструмент катары колдонот. SQUID серверин интернет трафиктин статистикалык маалыматын топтоодо да колдонуу мүмкүн. Колдонуучуларга контролдуу жол менен Интернет сервисине жол берүү, авторизациялоо, керексиз веб маалыматтарды фильтрлөө, жабуу мүмкүн. Ички коду ачык болгон бул сервер лицензиялоо жактан да абдан ыңгайлуу, жана бекер таратылгандыктан эң перспективдүү кэш сервери болуп келе жатат.

Азыркы күндөрдө бул сервердин функцияналдык мүмкүнчүлүктөрүн кеңейтүүдө көптөгөн эгемен жумуштар аткарылууда. Учурдагы изилдөөнүн негизинде сервер үчүн жардамчы кеңейтилөөлөрдүн кандайдыр бир умтулуу тенденциясы «access.log» лог файлынын тегерегинде жүрүүдө. Жана тилекке каршы «cache manager» протоколу көп учурда колдонулбагандыгы көрүнүүдө. Бул ички протокол жалаң гана SQUID сервердин иштетүүчүлөрү тарабынан гана колдонуулуда, жана көп учурда жалаң гана серверди тестирлөөдө колдонулуп жатат. Менин оюмча бул ички «cache manager» протоколу абдан бай жана маанилүү маалыматтарды камтыйт. Көп учурда туура эмес жол менен унутулуп калат. Керек болсо китептерде да бул протокол ыңгайсыз деп жазылып келе жатат: «The cache manager interface leaves much to be desired. It has a very unpolished feel. Novice administrators will probably find it difficult to use and understand. One of the first problems you might notice is that the menu (or table of contents) is unorganized. There is no logical order or grouping. The first items in the list provide low-level information primarily meant for developers. Currently, the order is determined by the initialization sequence in the source code.»[1]

Албетте бул протоколду абдан спецификалык деп айтуу болот. Жана биринчи көрүнүштө ыңгайсыз жана өтө тар деп айтуу болот. Бирок учурдагы изилдөөнүн негизинде бул протоколду колдонуу мүмкүндүгү жана керек болсо кеңейтүү, оңой колдонуу жолун иштетип чыгуу аракети жазалган жана көрсөтүлгөн.

Жогоруда келтирилген прототиптер бул протоколдун демонстрациялык мисалы катары келтирилген. Жана ар күн системдик администраторлордун көз алдында жана контрол алдында боло турган чоңдуктарды мониторлоого жардам берүүчү инструмент катары көрсөтүлдү. Интернет трафиктин мындайча дагы кеңири жол менен анализдене турганы жана ыңгайлуу түрдө жүргүзүү мүмкүнчүлүгү келтирилген.

Негизи, интернет трафиктин мониторинги келерки жылдарда абдан маанилүү болоору шексиз. Бул тенденция күндөн күнгө көбөйгөн сервистердин негизинен, бизнестин электрондук түргө өтүшү, керек болсо мамлекеттик сервистер да Интернетке келишинен өсө берет. Интернетте мониторинг жүргүзүү стратегиялык, коопсуздук жактан, интелектуалдык байлыкты коргоодо абдан маанилүү экендиги шексиз.

Библиогрфия

1. «Squid: The Definitive Guide» By Duane Wessels. ISBN 0-596-00162-2 January 2004
2. «Linux Security Cookbook» By Daniel J. Barrett, Robert G. Byrnes, Richard Silverman. ISDN 0-596-00391-9 June 2003
3. www.squid-cache.org
4. http://sourceforge.net/projects/free-sa/files/
5. http://sams.perm.ru/
6. http://lightsquid.sourceforge.net
7. http:// giannis.stoilis.gr/software/mysar/
8. http://sarg.sourceforge.net/
9. http://stc.nixdev.org
10. http://evc.fromru.com/squid2mysql/index.html
11. http://www.squidguard.org/
12. http://pb.pils.ru/
13. http://cyberos.narod.ru/statman/index.html
14. RFC 1413: Identification Protocol
15. RFC 1738: Uniform Resource Locators (URL)
16. RFC 2186: Internet Cache Protocol (ICP), version 2
17. RFC 2187: Application of Internet Cache Protocol (ICP), version 2
18. RFC 2396: Uniform Resource Identifiers (URI): Generic Syntax
19. RFC 2616: Hypertext Transfer Protocol -- HTTP/1.1
20. RFC 2617: HTTP Authentication: Basic and Digest Access Authentication
21. RFC 2756: Hypertext Caching Protocol
22. RFC 3040: Internet Web Replication and Caching Taxonomy
23. RFC 3143: Known HTTP Proxy/Caching Problems

ТИРКЕМЕ – 1

```
Array ( [server_version] =>
        squid/2.7.STABLE7
        [con] => Array ( [0x28367310] =>Array ( [peer] => 127.0.0.1:25064
        [me] => 127.0.0.1:3128
        [uri] => cache_object://localhost/active_requests
        [bytes] => 0
        [seconds] => 0
        [username] => - )
    [0x28367410] => Array ( [peer] => 192.168.20.22:3167
        [me] => 192.168.20.222:3128
        [uri] => http://rs.mail.ru/b10275160.jpg
        [bytes] => 0
        [seconds] => 0
        [username] => - )
    [0x28367110] => Array ( [peer] => 192.168.20.22:316
        [me] => 192.168.20.222:3128
        [uri] => http://rs.mail.ru/b10221756.jpg
        [bytes] => 2999
        [seconds] => 0
        [username] => - )
    [0x28367510] => Array ( [peer] => 192.168.20.22:3171
        [me] => 192.168.20.222:3128
        [uri] => http://rs.mail.ru/b10276305.jpg
        [bytes] => 0
        [seconds] => 0
        [username] => - )
    [0x28367210] => Array ( [peer] => 192.168.20.22:3152
        [me] => 192.168.20.222:3128
        [uri] => http://img.mail.ru/r/js/blogs/tooltiplib.js
        [bytes] => 0
        [seconds] => 0
```

[username] => -)

[0x28367010] => Array ([peer] => 192.168.20.22:3138

[me] => 192.168.20.222:3128

[uri] => http://img.mail.ru/0.gif

[bytes] => 0

[seconds] => 0

[username] => -)))

**********DISKD бөлүмү**********

Array ([0] => HTTP/1.0 200 OK

[1] => Server: squid/2.7.STABLE7

[2] => Date: Tue, 11 May 2010 09:53:02 GMT

[3] => Content-Type: text/plain

[4] => Expires: Tue, 11 May 2010 09:53:02 GMT

[5] => X-Cache: MISS from manas.kg

[6] => Via: 1.0 manas.kg:3128 (squid/2.7.STABLE7)

[7] => Connection: close

[8] =>

[9] => sent_count: 0

[10] => recv_count: 0

[11] => max_away: 0

[12] => max_shmuse: 0

[13] => open_fail_queue_len: 0

[14] => block_queue_len: 0

[15] =>

[16] => OPS SUCCESS FAIL

[17] => open 0 0 0

[18] => create 0 0 0

[19] => close 0 0 0

[20] => unlink 0 0 0

[21] => read 0 0 0

[22] => write 0 0 0

[23] =>)

**********IPCACHE Бөлүмү**********

Array ([0] => HTTP/1.0 200 OK

[1] => Server: squid/2.7.STABLE7

[2] => Date: Tue, 11 May 2010 09:57:17 GMT

[3] => Content-Type: text/plain

[4] => Expires: Tue, 11 May 2010 09:57:17 GMT

[5] => X-Cache: MISS from manas.kg

[6] => Via: 1.0 manas.kg:3128 (squid/2.7.STABLE7)

[7] => Connection: close

[8] =>

[9] => IP Cache Statistics:

[10] => IPcache Entries: 14

[11] => IPcache Requests: 41

[12] => IPcache Hits: 21

[13] => IPcache Negative Hits: 0

[14] => IPcache Numeric Hits: 0

[15] => IPcache Misses: 20

[16] => IPcache Invalid Requests: 0

[17] =>

[18] =>

[19] => IP Cache Contents:

[20] =>

[21] => Hostname Flg lstref TTL N

[22] => safebrowsing-cache.google.com 417 -311 1(0) 74.125.106.90-OK

[23] => safebrowsing.clients.google.com 418 -173 6(0) 74.125.39.102-OK 74.125.39.101-OK 74.125.39.139-OK 74.125.39.113-OK 74.125.39.100-OK 74.125.39.138-OK

[24] => clients4.google.com 625 -345 6(0) 74.125.39.138-OK 74.125.39.113-OK 74.125.39.102-OK 74.125.39.139-OK 74.125.39.101-OK 74.125.39.100-OK

[25] => rs.mail.ru 651 -600 1(0) 94.100.179.178-OK

[26] => img.mail.ru 653 -595 1(0) 94.100.189.60-OK

[27] => mail.radar.imgsmail.ru 654 -601 1(0) 94.100.182.110-OK

[28] => top5.mail.ru 654 -594 1(0) 94.100.185.25-OK

[29] => img.imgsmail.ru 655 -601 1(0) 94.100.189.180-OK

[30] => auth.mail.ru 660 2935 1(0) 217.69.128.11-OK

[31] => www.tns-counter.ru 660 2915 5(0) 217.73.200.222-OK 217.73.200.169-OK 217.73.200.219-OK 217.73.200.220-OK 217.73.200.221-OK

[32] => mail.ru 663 -603 5(0) 217.69.128.41-OK 217.69.128.42-OK 217.69.128.43-OK 217.69.128.44-OK 217.69.128.45-OK

[33] => manas.kg H 953 -1 1(0) 192.168.20.241-OK

[34] => localhost.my.domain H 953 -1 1(0) 127.0.0.1-OK

[35] => localhost H 953 -1 1(0) 127.0.0.1-OK

[36] =>)

**********INFO Бөлүмү**********

Array ([0] => HTTP/1.0 200 OK

[1] => Server: squid/2.7.STABLE7

[2] => Date: Tue, 11 May 2010 11:04:53 GMT

[3] => Content-Type: text/plain

[4] => Expires: Tue, 11 May 2010 11:04:53 GMT

[5] => X-Cache: MISS from manas.kg

[6] => Via: 1.0 manas.kg:3128 (squid/2.7.STABLE7)

[7] => Connection: close

[8] =>

[9] => Squid Object Cache: Version 2.7.STABLE7

[10] => Start Time: Tue, 11 May 2010 09:41:24 GMT

[11] => Current Time: Tue, 11 May 2010 11:04:53 GMT

[12] => Connection information for squid:

[13] => Number of clients accessing cache: 2

[14] => Number of HTTP requests received: 140

[15] => Number of ICP messages received: 0

[16] => Number of ICP messages sent: 0

[17] => Number of queued ICP replies: 0

[18] => Request failure ratio: 0.00

[19] => Average HTTP requests per minute since start: 1.7

[20] => Average ICP messages per minute since start: 0.0

[21] => Select loop called: 10353 times, 483.852 ms avg

[22] => Cache information for squid:

[23] => Request Hit Ratios: 5min: 0.0%, 60min: 0.0%

[24] => Byte Hit Ratios: 5min: -0.0%, 60min: 5.9%

[25] => Request Memory Hit Ratios: 5min: 0.0%, 60min: 0.0%

[26] => Request Disk Hit Ratios: 5min: 0.0%, 60min: 0.0%

[27] => Storage Swap size: 22000 KB

[28] => Storage Mem size: 560 KB

[29] => Mean Object Size: 10.52 KB

[30] => Requests given to unlinkd: 67

[31] => Median Service Times (seconds) 5 min 60 min:

[32] => HTTP Requests (All): 0.00000 0.25890

[33] => Cache Misses: 0.00000 0.25890

[34] => Cache Hits: 0.00000 0.00000

[35] => Near Hits: 0.00000 0.00000

[36] => Not-Modified Replies: 0.00000 0.00000

[37] => DNS Lookups: 0.00000 0.00203

[38] => ICP Queries: 0.00000 0.00000

[39] => Resource usage for squid:

[40] => UP Time: 5009.318 seconds

[41] => CPU Time: 1.652 seconds

[42] => CPU Usage: 0.03%

[43] => CPU Usage, 5 minute avg: 0.01%

[44] => CPU Usage, 60 minute avg: 0.02%

[45] => Process Data Segment Size via sbrk(): 840 KB
[46] => Maximum Resident Size: 6672 KB
[47] => Page faults with physical i/o: 2
[48] => Memory accounted for:
[49] => Total accounted: 1058 KB
[50] => memPoolAlloc calls: 47599
[51] => memPoolFree calls: 38001
[52] => File descriptor usage for squid:
[53] => Maximum number of file descriptors: 3520
[54] => Largest file desc currently in use: 14
[55] => Number of file desc currently in use: 11
[56] => Files queued for open: 0
[57] => Available number of file descriptors: 3509
[58] => Reserved number of file descriptors: 100
[59] => Store Disk files open: 0
[60] => IO loop method: kqueue
[61] => Internal Data Structures:
[62] => 2117 StoreEntries
[63] => 125 StoreEntries with MemObjects
[64] => 124 Hot Object Cache Items
[65] => 2091 on-disk objects
[66])

ТИРКЕМЕ – 2

Жардамчы класс файл

```
<?php

// sqstat class

class squidstat{
      var $fp;
      var $errstr;
      var $errno;

      var $use_sessions=false;
      function squidstat(){
            if (!function_exists("preg_match")){
                  $this->errno=5;
                  $this->errstr='You need to install <a
href="http://www.php.net/pcre/" target="_blank">PHP pcre
extension</a> to run this script';
                  $this->showError();
                  exit(5);
            }

            // we need session support to gather avg. speed
            if (function_exists("session_start")){
                  $this->use_sessions=true;
            }

      }

      function formatXHTML($body,$refresh,$use_js=false){
            $text='<?xml version="1.0" encoding="UTF-8"?>'."\n".
            '<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0
Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-
transitional.dtd">'."\n"
            .'<html>'
            .'<head>'
```

```
            .'<link href="sqstat.css" rel="stylesheet"
type="text/css"/>';
            if($refresh) $text.='<META HTTP-EQUIV=Refresh
CONTENT="'.$refresh.';
URL='.$_SERVER["PHP_SELF"].'?refresh='.$refresh.'&config='.$GLOBALS[
"config"].'"/>';
            $text.='<title>Squidтин заматта мониторинги</title>'
            .($use_js?'<script src="zhabascript.js"
type="text/javascript"></script>':'').'</head>'
            .($use_js?'<body onload="jsInit();"><div
id="dhtmltooltip"></div><img id="dhtmlpointer"
src="arrow.gif">':'<body>')
            .'<ul id="menu">'
            .'<li><a href="sqstat.php">Суроо-Талаптар
мониторинги</a></li>'
            .'<li><a href="index.php">Жалпы маалымат</a></li>'
            .'<li><a href="filedesc.php">Файл
дескрипторлор</a></li>'
            .'<li><a href="storedir.php">Кэш статистикасы</a></li>'
            .'</ul>'
            .$body.'</body></html>';
            return $text;
      }

      function showError(){
            $text='<h1>SqStat error</h1>'.
            '<h2 style="color:red">Error ('.$this->errno.'):
'.$this->errstr.'</span>';
            echo $this->formatXHTML($text,0);
      }
      function connect($squidhost,$squidport){
            $this->fp = false;
            // Connecting to the squidhost
            $this->fp = @fsockopen($squidhost, $squidport, $this-
>errno, $this->errstr, 10);
            if (!$this->fp) {
                  // failed to connect
                  return false;
            }
```

```
            return true;
      }
      function duration ($seconds) {
            $takes_time = array(604800,86400,3600,60,0);
            $suffixes = array("w","d","h","m","s");
            $output = "";
            foreach ($takes_time as $key=>$val) {
                  ${$suffixes[$key]} = ($val == 0) ? $seconds :
floor(($seconds/$val));
                  $seconds -= ${$suffixes[$key]} * $val;
                  if (${$suffixes[$key]} > 0) {
                        $output .=  ${$suffixes[$key]};
                        $output .= $suffixes[$key]." ";
                  }
            }
            return trim($output);
      }

      function filesize_format($bytes, $format = '', $force = '')
      {
            $force = strtoupper($force);
            $defaultFormat = '%01d %s';
            if (strlen($format) == 0)
            $format = $defaultFormat;
            $bytes = max(0, (int) $bytes);
            $units = array('b', 'Kb', 'Mb', 'Gb', 'Tb', 'Pb');
            $power = array_search($force, $units);
            if ($power === false)
            $power = $bytes > 0 ? floor(log($bytes)/log(1024)) : 0;
            return sprintf($format, $bytes / pow(1024, $power),
$units[$power]);
      }
      function makeStoreDirQuery($pass=''){
            $raw=array();
            // sending request
            if(!$this->fp) die("Please connect to server");
            $out = "GET cache_object://localhost/storedir
HTTP/1.0\r\n";
```

```
            if($pass!="") $out.="Authorization: Basic
".base64_encode("cachemgr:$pass")."\r\n";
            $out.="\r\n";
            fwrite($this->fp, $out);

            while (!feof($this->fp)) {
                  $raw[]=trim(fgets($this->fp, 2048));
            }
            fclose($this->fp);

            if($raw[0]!="HTTP/1.0 200 OK"){
                  $this->errno=1;
                  $this->errstr="Cannot get data. Server answered:
$raw[0]";
                  return false;
            }


            //print_r($raw);
            $header=1;
            $connection=0;
            $parsed["Сервер версиясы"]="Unknown";
            foreach($raw as $key=>$v){
                  // cutoff http header
                  if($header==1 && $v=="") $header=0;
                  if($header){
                        if(substr(strtolower($v),0,7)=="server:"){
// parsing server version
                              $parsed["Сервер
версиясы"]=substr($v,8);
                        }
                  }
                  else {
                              /* username field is avaible in Squid
2.6 stable */

                              if(substr($v,0,13)=="Store Entries")
$parsed["Кэштеги обьктилердин саны"]=substr($v,strpos($v, ":")+1);
```

```
                              if(substr($v,0,17)=="Maximum Swap
Size") $parsed["Кэштин максимум эс колому"]=substr($v,strpos($v,
":")+1);
                              if(substr($v,0,23)=="Current Store
Swap Size") $parsed["Учурдагы кэштин колому"]=substr($v,strpos($v,
":")+1);
                              if(substr($v,0,16)=="Current
Capacity") $parsed["Учурдагы кэтин колому процент
менен"]=substr($v,strpos($v, ":")+1);

                        }
            }
            return $parsed;
      }
      function makeFileDescQuery($pass=''){

            $raw=array();
            // sending request
            if(!$this->fp) die("Please connect to server");
            $out = "GET cache_object://localhost/filedescriptors
HTTP/1.0\r\n";
            if($pass!="") $out.="Authorization: Basic
".base64_encode("cachemgr:$pass")."\r\n";
            $out.="\r\n";
            fwrite($this->fp, $out);

            while (!feof($this->fp)) {
                  $raw[]=trim(fgets($this->fp, 2048));
            }
            fclose($this->fp);

            if($raw[0]!="HTTP/1.0 200 OK"){
                  $this->errno=1;
                  $this->errstr="Cannot get data. Server answered:
$raw[0]";
                  return false;
            }
```

```
//          print_r($raw);
//          exit;
            $header=1;
            $bheader=1;
            $connection=0;
            $parsed["Сервер версиясы"]="Unknown";
            $tdata = "<br />";
            foreach($raw as $key=>$v){
                  // cutoff http header
                  if($header==1 && $v=="") $header=0;
                  if($header){
                        if(substr(strtolower($v),0,7)=="server:"){
// parsing server version
                              $parsed["Сервер
версиясы"]=substr($v,8);
                        }
                  }

                  if($bheader==1 && substr($v,0,4)=="----")
{$bheader=0; continue;}
                  if(!$bheader)
                        {
                              /* username field is avaible in Squid
2.6 stable */
                        //list($file, $type, $tout, $nread, $nwrite,
$remaddr, $desc, $bulk) = explode(" ", $v);
                        //$tdata = $tdata."<tr><td>$file</td>
<td>$type</td> <td>$tout</td> <td>$nread</td> <td>$nwrite</td>
<td>$remaddr $desc $bulk</td></tr>";
                        $tdata = $tdata."<tr><td>".$v."</td></tr>";

                        }
            }
            $text ='<div class="header"><form method="get"
action="'.$_SERVER["PHP_SELF"].'">'.
            'Squidтин заматта мониторинги. Сервер версиясы
('.$parsed["Сервер версиясы"].').<br/>'.
            'Жанылоо убактысы: <input name="refresh" type="text"
size="4" value="'.($_GET['refresh']?$_GET['refresh']:0).'"/> sec.
```

```
<input type="submit" value="Жаныла"/> <input name="stop"
type="submit" value="Токтот"/> Баштоо убактысы: <tt>'.date("h:i:s
d/m/Y").'</tt><br/>'.
            '</div>'.
            "<table>".
            "<th> <td>Адресс</td></th>"
            .$tdata.
            "</table>";
            return $this-
>formatXHTML($text,$_GET['refresh']?$_GET['refresh']:0);

      }


      function makeInfoQuery($pass=''){
            $raw=array();
            // sending request
            if(!$this->fp) die("Please connect to server");
            $out = "GET cache_object://localhost/info HTTP/1.0\r\n";
            if($pass!="") $out.="Authorization: Basic
".base64_encode("cachemgr:$pass")."\r\n";
            $out.="\r\n";
            fwrite($this->fp, $out);

            while (!feof($this->fp)) {
                  $raw[]=trim(fgets($this->fp, 2048));
            }
            fclose($this->fp);

            if($raw[0]!="HTTP/1.0 200 OK"){
                  $this->errno=1;
                  $this->errstr="Cannot get data. Server answered:
$raw[0]";
                  return false;
            }


            //print_r($raw);
            $header=1;
```

```
            $connection=0;
            $parsed["Сервер версиясы"]="Unknown";
            foreach($raw as $key=>$v){
                  // cutoff http header
                  if($header==1 && $v=="") $header=0;
                  if($header){
                        if(substr(strtolower($v),0,7)=="server:"){
// parsing server version
                              $parsed["Сервер
версиясы"]=substr($v,8);
                        }
                  }
                  else {
                              /* username field is avaible in Squid
2.6 stable */

                              if(substr($v,0,11)=="Start Time:")
$parsed["Сервердин баштоо убактыты"]=substr($v,11);
                              if(substr($v,0,13)=="Current Time:")
$parsed["Учурдагы сервер убакыты"]=substr($v,13);
                              if(substr($v,0,34)=="Number of clients
accessing cache:") $parsed["Колдонуучулар саны"]=substr($v,34);
                              if(substr($v,0,33)=="Number of HTTP
requests received:") $parsed["http суроо-талаптар"]=substr($v,33);
                              if(substr($v,0,45)=="Average HTTP
requests per minute since start:") $parsed["Минутадагы орточо http
с.т"]=substr($v,45);
                              if(substr($v,0,18)=="Storage Swap
size:") $parsed["Катуу дисктеги Swape колому (кб)"]=substr($v,18);
                              if(substr($v,0,9)=="CPU Time:")
$parsed["CPU убактысы"]=substr($v,9);
                              if(substr($v,0,10)=="CPU Usage:")
$parsed["CPU колдонуусу"]=substr($v,10);
                              if(substr($v,0,24)=="CPU Usage, 5
minute avg:") $parsed["CPU 5 минута ичинде
колдонуусу"]=substr($v,24);
                              if(substr($v,0,25)=="CPU Usage, 60
minute avg:") $parsed["CPU 60 минута ичинде
колдонуусу"]=substr($v,25);
```

```
                              if(substr($v,0,21)=="Total space in
arena:") $parsed["Оперативдик эс колому"]=substr($v,21);
                              if(substr($v,0,11)=="Total free:")
$parsed["Бош оперативлик эс колому"]=substr($v,11);
                              if(substr($v,0,35)=="Maximum number of
file descriptors:") $parsed["Максимум файл дескрипторлордун
саны"]=substr($v,35);
                              if(substr($v,0,37)=="Number of file
desc currently in use:") $parsed["Учурдагы колд. файл дескр
саны"]=substr($v,37);
                              if(substr($v,0,37)=="Available number
of file descriptors:") $parsed["Бош файл дескр саны"]=substr($v,37);

                        }
            }
            return $parsed;
      }
      function makeHtmlInfoReport($data){
      $refresh=0;
            if(isset($_GET["refresh"]) && !isset($_GET["stop"]))
$refresh=(int)$_GET["refresh"];
            $text='';
            if(count($GLOBALS["configs"])==1)
$servers=$GLOBALS["squidhost"].':'.$GLOBALS["squidport"];
            else{
                  $servers='<select onchange="this.form.submit();"
name="config">';
                  foreach ($GLOBALS["configs"] as $key=>$v){
                        $servers.='<option
'.($GLOBALS["config"]==$key?' selected="selected" ':'').'
value="'.$key.'">'.htmlspecialchars($v).'</option>';
                  }
                  $servers.='</select>';
            }
       $text.='<div class="header"><form method="get"
action="'.$_SERVER["PHP_SELF"].'">'.
            'Squidтин заматта мониторинги. Сервер адреси'.$servers.'
версиясы ('.$data["Сервер версиясы"].').<br/>'.
```

```
            'Жанылоо убактысы: <input name="refresh" type="text"
size="4" value="'.$refresh.'"/> sec. <input type="submit"
value="Жаныла"/> <input name="stop" type="submit" value="Токтот"/>
Баштоо убактысы: <tt>'.date("h:i:s d/m/Y").'</tt><br/>'.
            '</div>'.
            '<table class="result" align="center" width="100%"
border="0">'.
            '<tr>'.
            '<th>Касиет</th><th>Маанизи</th>';

      foreach($data as $key=>$value){
            $text.="<tr><td>$key</td><td>$value</td></tr>";
      }
      $text.="</table>";
      return $this->formatXHTML($text,$refresh);

      }
      function makeQuery($pass=""){
            $raw=array();
            // sending request
            if(!$this->fp) die("Please connect to server");
            $out = "GET cache_object://localhost/active_requests
HTTP/1.0\r\n";
            if($pass!="") $out.="Authorization: Basic
".base64_encode("cachemgr:$pass")."\r\n";
            $out.="\r\n";
            fwrite($this->fp, $out);

            while (!feof($this->fp)) {
                  $raw[]=trim(fgets($this->fp, 2048));
            }
            fclose($this->fp);

            if($raw[0]!="HTTP/1.0 200 OK"){
                  $this->errno=1;
                  $this->errstr="Cannot get data. Server answered:
$raw[0]";
                  return false;
            }
```

```
            // parsing output;
            $header=1;
            $connection=0;
            $parsed["Сервер версиясы"]="Unknown";
            foreach($raw as $key=>$v){
                  // cutoff http header
                  if($header==1 && $v=="") $header=0;
                  if($header){
                        if(substr(strtolower($v),0,7)=="server:"){
// parsing server version
                              $parsed["Сервер
версиясы"]=substr($v,8);
                        }
                  }
                  else {
                        if(substr($v,0,11)=="Connection:"){ //
parsing connection
                              $connection=substr($v,12);
                        }
                        if($connection){
                              /* username field is avaible in Squid
2.6 stable */
                              if(substr($v,0,9)=="username ")
$parsed["con"][$connection]["username"]=substr($v,9);
                              if(substr($v,0,5)=="peer:")
$parsed["con"][$connection]["peer"]=substr($v,6);
                              if(substr($v,0,3)=="me:")
$parsed["con"][$connection]["me"]=substr($v,4);
                              if(substr($v,0,4)=="uri ")
$parsed["con"][$connection]["uri"]=substr($v,4);
                              if(substr($v,0,10)=="delay_pool")
$parsed["con"][$connection]["delay_pool"]=substr($v,11);

                              if(preg_match('/out.offset \d+,
out.size (\d+)/',$v,$matches)){

      $parsed["con"][$connection]["bytes"]=$matches[1];
                              }
```

```
                              if(preg_match('/start \d+\.\d+
\((\d+).\d+ seconds ago\)/',$v,$matches)){

      $parsed["con"][$connection]["seconds"]=$matches[1];
                              }
                        }
                  }
            }
            return $parsed;
      }
      function implode_with_keys($array, $glue) {
            foreach ($array as $key=>$v){
                  $ret[]=$key.'='.htmlspecialchars($v);
            }
            return implode($glue, $ret);
      }
      function
makeHtmlReport($data,$resolveip=false,$hosts_array=array(),$use_js=t
rue){
            global $group_by;
            if($this->use_sessions){
                  session_name('SQDATA');
                  session_start();
            }

            $total_avg = $total_curr = 0;
            // resort data array
            $users=array();
            switch($group_by){
                  case "host":
                  $group_by_name="Host";
                  $group_by_key='return $ip;';
                  break;
                  case "username":
                  $group_by_name="User";
                  $group_by_key='return $v["username"];';
                  break;
                  default:
                  die("wrong group_by!");
```

```
            }

            foreach($data["con"] as $key => $v){
                  if(substr($v["uri"],0,13)=="cache_object:")
continue; // skip myself
                  $ip=substr($v["peer"],0,strpos($v["peer"],":"));
                  if(isset($hosts_array[$ip])){
                        $ip=$hosts_array[$ip];
                  }
                  // i use ip2long() to make ip sorting work
correctly
                  elseif($resolveip){
                        $hostname=gethostbyaddr($ip);
                        if($hostname==$ip) $ip=ip2long($ip);//
resolve failed
                        else $ip=$hostname;
                  }
                  else{

      $ip=ip2long(substr($v["peer"],0,strpos($v["peer"],":")));
                  }
                  $v['connection'] = $key;
                  if(!isset($v["username"])) $v["username"]="N/A";
                  $users[eval($group_by_key)][]=$v;
            }
            ksort($users);
            $refresh=0;
            if(isset($_GET["refresh"]) && !isset($_GET["stop"]))
$refresh=(int)$_GET["refresh"];
            $text='';
            if(count($GLOBALS["configs"])==1)
$servers=$GLOBALS["squidhost"].':'.$GLOBALS["squidport"];
            else{
                  $servers='<select onchange="this.form.submit();"
name="config">';
                  foreach ($GLOBALS["configs"] as $key=>$v){
                        $servers.='<option
'.($GLOBALS["config"]==$key?' selected="selected" ':'').'
value="'.$key.'">'.htmlspecialchars($v).'</option>';
```

```
                }
                $servers.='</select>';
            }
            $text.='<div class="header"><form method="get"
action="'.$_SERVER["PHP_SELF"].'">'.
            'Squidтин заматта мониторинги. Сервер адреси'.$servers.'
версиясы ('.$data["Сервер версиясы"].').<br/>'.
            'Жанылоо убактысы: <input name="refresh" type="text"
size="4" value="'.$refresh.'"/> sec. <input type="submit"
value="Жаныла"/> <input name="stop" type="submit" value="Токтот"/>
Баштоо убактысы: <tt>'.date("h:i:s d/m/Y").'</tt><br/>'.
            '</div>'.
            '<table class="result" align="center" width="100%"
border="0">'.
            '<tr>'.
            '<th>'.$group_by_name.'</th><th>URI</th>'.
            ($this-
>use_sessions?'<th>учур.ылд</th><th>орт.ылд</th>':'').
            '<th>колом</th><th>убакыт</th>'.
            '</tr>';
            $ausers=$acon=0;
            unset($session_data);
            if (isset($_SESSION['time']) && ((time() -
$_SESSION['time']) < 3*60) && isset($_SESSION['sqdata']) &&
is_array($_SESSION['sqdata'])) {
                //only if the latest data was less than 3 minutes
ago
                $session_data = $_SESSION['sqdata'];
            }
            $table='';
            foreach($users as $key=>$v){
                $ausers++;
                $table.='<tr><td style="border-right:0;"
colspan="2"><b>'.(is_int($key)?long2ip($key):$key).'</b></td>'.
                '<td style="border-left:0;"
colspan="5"> </td></tr>';
                $user_avg = $user_curr = $con_color =  0;

                foreach ($v as $con){
```

```
                        if(substr($con["uri"],0,7)=="http://" ||
substr($con["uri"],0,6)=="ftp://"){
                                if(strlen($con["uri"])>SQSTAT_SHOWLEN)
$uritext=htmlspecialchars(substr($con["uri"],0,SQSTAT_SHOWLEN)).'</a
> ....';
                                else
$uritext=htmlspecialchars($con["uri"]).'</a>';
                                $uri='<a target="_blank"
href="'.htmlspecialchars($con["uri"]).'">'.$uritext;
                        }
                        else $uri=htmlspecialchars($con["uri"]);
                        $acon++;
                        //speed stuff
                        $con_id = $con['connection'];
                        $is_time = time();
                        $curr_speed=0;
                        $avg_speed=0;
                        if (isset($session_data[$con_id]) &&
$con_data = $session_data[$con_id] ) {
                                // if we have info about current
connection, we do analyze its data
                                // current speed
                                $was_time = $con_data['time'];
                                $was_size = $con_data['size'];
                                if ($was_time && $was_size) {
                                      $delta = $is_time - $was_time;
                                      if ($delta == 0) {
                                            $delta = 1;
                                      }
                                      if ($con['bytes'] >= $was_size) {
                                            $curr_speed =
($con['bytes'] - $was_size) / 1024 / $delta;
                                      }
                                } else {
                                      $curr_speed = $con['bytes'] /
1024;
                                }

                                //avg speed
```

```
                        $avg_speed = $con['bytes'] / 1024;
                        if ($con['seconds'] > 0) {
                              $avg_speed /= $con['seconds'];
                        }
                  }

                  $new_data[$con_id]['time'] = $is_time;
                  $new_data[$con_id]['size'] = $con['bytes'];

                  //sum speeds
                  $total_avg += $avg_speed;
                  $user_avg += $avg_speed;
                  $total_curr += $curr_speed;
                  $user_curr += $curr_speed;

                  if($use_js)
$js='onMouseout="hideddrivetip()" onMouseover="ddrivetip(\''.$this-
>implode_with_keys($con,'<br/>').'\')"';
                  else $js='';
                  $table.='<tr'.( (++$con_color % 2 == 0) ? '
class="odd"' : '' ).'><td id="white"></td>'.
                  '<td nowrap '.$js.' width="80%"
>'.$uri.'</td>';
                  if($this->use_sessions){
                        $table .= '<td nowrap
align="right">'.( (round($curr_speed, 2) > 0) ? sprintf("%01.2f
KB/s", $curr_speed) : '' ).'</td>'.
                        '<td nowrap align="right">'.(
(round($avg_speed, 2) > 0) ? sprintf("%01.2f KB/s", $avg_speed) : ''
). '</td>';
                  }
                  $table .= '<td nowrap align="right">'.$this-
>filesize_format($con["bytes"]).'</td>'.
                  '<td nowrap align="right">'.$this-
>duration($con["seconds"],"short").'</td>'.
                  '</tr>';
            }
            if($this->use_sessions){
```

```
                        $table.=sprintf("<tr><td
colspan=\"2\"></td><td align=\"right\" id=\"highlight\">%01.2f
KB/s</td><td align=\"right\" id=\"highlight\">%01.2f KB/s</td><td
colspan=\"2\"></td>",
                        $user_curr, $user_avg);
                  }

            }
            $_SESSION['time'] = time();
            if(isset($new_data)) $_SESSION['sqdata'] = $new_data;
            $stat_row='';
            if($this->use_sessions){
                  $stat_row.=sprintf("<tr
class=\"total\"><td><b>Баардыгы:</b></td><td align=\"right\"
colspan=\"5\"><b>%d</b> колдонуучу жана <b>%d</b> байланыш @
<b>%01.2f/%01.2f</b> KB/s (CURR/AVG)</td></tr>",
                  $ausers, $acon, $total_curr, $total_avg);
            }
            else {
                  $stat_row.=sprintf("<tr
class=\"total\"><td><b>Баардыгы:</b></td><td align=\"right\"
colspan=\"5\"><b>%d</b> колдонуучу жана <b>%d</b>
байланыш</td></tr>",
                  $ausers, $acon);
            }
            if($ausers==0){
                  $text.='<tr><td colspan=6><b>Байланыш
жок</b></td></tr>';
            }
            else {
                  $text.=$stat_row.$table.$stat_row;
            }
            $text .= '</table>'.
            '';
            return $this->formatXHTML($text,$refresh,$use_js);
      }
}
?>
```

Squid прокси сервер негизинде веб трафиктин онлайн мониторинги жана эсеп системи.

(МАГИСТР ДИПЛОМУ)

Тилек Майтыков

БИШКЕК 2010

КЫРГЫЗ-ТҮРК МАНАС УНИВЕРСИТЕТИ

ТАБИГЫЙ ИЛИМДЕР ИНСТИТУТУ

КОМПЬЮТЕР ИНЖЕНЕРИЯ БАГЫТЫ

Squid прокси сервер негизинде веб трафиктин онлайн мониторинги жана эсеп системи.

(Squid proxy server bazında web trafiginin online gözlenmesi ve hesab sistemi.)

(МАГИСТР ДИПЛОМУ)

Тилек Майтыков

БИШКЕК 2010