

# Implementasi otentikasi pada squid dalam mode transparent proxy

Oleh:

Henry Saptono <[boypyt@gmail.com](mailto:boypyt@gmail.com)>

Mei 2010

Anda ingin implementasi otentikasi (*authentication*) pada squid proxy server ? Jika ya, sangat mudah karena squid telah dilengkapi dengan berbagai modul/program bantu untuk otentikasi. Dan mungkin Anda dapat menerapkannya dengan panduan dari berbagai artikel atau tutorial di internet ataupun dari artikel-artikel yang pernah penulis buat sebelumnya tentang *otentikasi squid*. Namun yang menjadi pertanyaan besar adalah bagaimana jika ingin menerapkan proses *otentikasi* pada proxy server squid dalam mode **transparent proxy** ?. Pada prinsipnya dalam mode transparent proxy kita tidak dapat menggunakan metode otentikasi dikarenakan masalah dalam TCP/IP ketika proses *port redirection* dari 80 ke 3128 (*squid default port*) bukan dikarenakan masalah pada squid itu sendiri. **Lalu** bagaimanakah caranya agar kita **masih dapat menerapkan metode otentikasi** dalam mode transparent proxy? Salah satu jawabannya adalah dengan memanfaatkan fitur **url rewrite** pada squid (versi > = 2.6).

Untuk itu pada artikel kali ini penulis akan mencoba memberikan contoh penggunaan fitur **url rewrite** yang tersedia pada squid proxy server untuk implementasi otentikasi pada squid proxy server yang berjalan dalam mode transparent.

Penulis dalam pembahasan artikel kali ini menggunakan sistem linux CentOS 5, dan menggunakan squid bawaan distro CentOS 5, serta menggunakan php cli (*php commandline interface*) untuk mengeksekusi program atau script yang akan melakukan *url rewrite* terhadap *url request* dari komputer klien.

Penulis juga akan membuat program untuk *url rewrite* menggunakan bahasa pemrograman php dan beberapa script php untuk aplikasi login, database yang digunakan adalah database mysql, serta tentunya membutuhkan apache web server.

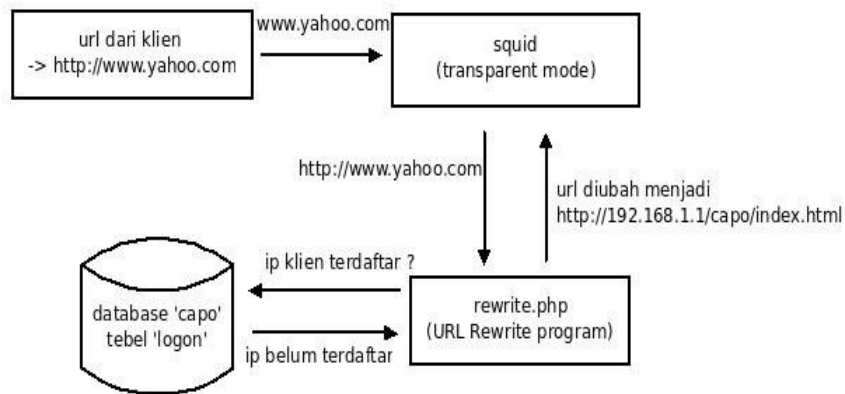
Script atau program url rewrite yang penulis buat dalam contoh di artikel ini belum menekankan pada aspek pemrograman yang baik dan aman namun sebatas pada fungsi yang dapat dilakukan oleh script atau program tersebut sebagai pemroses *url rewrite* untuk memungkinkan terjadinya proses otentikasi pada squid meskipun berjalan dalam mode transparent. Metode otentikasi yang dilakukan melalui script ini juga masih sederhana sehingga mungkin bagi para programmer yang biasa dalam membuat program akan terlihat sangat kurang, hal ini dikarenakan penulis hanya menekankan pada kegunaan dan manfaat dari fitur url rewrite pada squid.

## Skenario

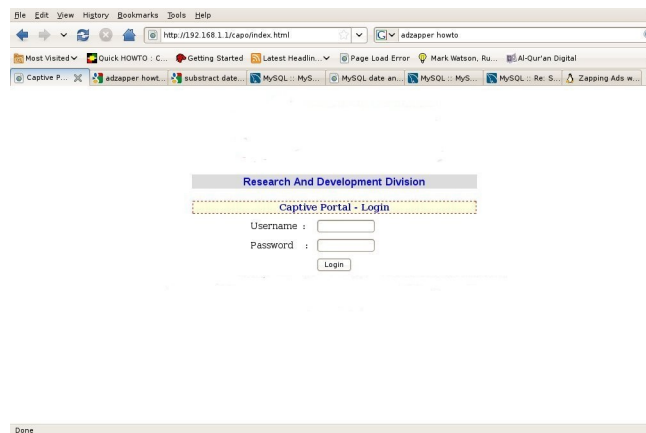
Untuk memudahkan penjelasan dan memberikan kepehaman kepada pembaca, penulis akan menjelaskan skenario implementasi otentikasi pada squid server dalam mode transparent, sebagai berikut:

- Setiap request akses web dari komputer-komputer klien oleh komputer gateway akan di *redirect* ke squid proxy server yang berjalan dalam mode transparent. Dalam contoh ini komputer gateway juga berperan sebagai squid proxy server.
- Komputer gateway memiliki dua buah network interface yaitu eth0 dan eth1. Inteface eth0 terhubung dengan modem/router, dan interface eth1 terhubung dengan switch LAN (192.168.1.0/24). IP eth1 adalah 192.168.1.1
- Setiap url (alamat web) tujuan yang diakses oleh klien (user) oleh squid proxy server akan di *redirect* ke sebuah program atau script yang dibuat dengan bahasa pemrograman php, yang diberi nama *rewrite.php* (file ini dibuat dan diletakkan kedalam direktori /usr/lib/squid/). Program *rewrite.php* terinstal pada komputer yang sama yaitu pada komputer gateway/squid proxy server.
- Script *rewrite.php* akan memproses setiap *url request* dengan algoritma sebagai berikut:
  - Jika alamat IP komputer klien belum terdaftar pada tabel '*logon*' pada database mysql yang diberi nama '*capo*' maka url dari klien akan dimanipulasi/diubah sehingga url nya menuju ke halaman aplikasi login (lihat gambar-1) yang dibuat dengan bahasa pemrograman php dan diinstal pada komputer gateway/proxy server. User harus login terlebih dahulu menggunakan user account yang sudah ada pada tabel '*user*' agar data alamat IP komputer klien didaftar/dicatat kedalam tabel *logon*. Setelah itu barulah user dapat mengakses web lainnya. Halaman login tampak seperti pada gambar-2.
  - Jika alamat IP komputer klien sudah terdaftar tabel '*logon*' maka user dapat mengakses web yang dituju, dalam hal ini alamat web site tujuan tidak diubah.
- Sekali user/klien sudah melalui proses otentikasi maka komputer user/klien bisa mengakses web apapun tanpa batasan
- Agar dihari lainnya user/klien harus melalui proses otentikasi lagi ketika baru memulai akses web maka data daftar ip komputer klien yang ada pada tabel *logon* setiap harinya harus dihapus pada setiap pergantian hari, misalnya dalam contoh ini penulis memanfaatkan service *crond* untuk melakukan penghapusan data IP komputer klien pada tabel *logon* secara kontinyu setiap hari di jam 01:05 dinihari. *Meskipun sebenarnya Anda dapat membuat script atau program *rewrite.php* lebih smart lagi misalnya menyertakan dukungan pengecekan durasi atau quota waktu dan lain lain..*

### Algoritma url rewrite jika IP klien belum terdaftar pada tabel logon



gambar-1. Algoritma proses url rewrite



Gambar-2. Halaman login

### Konfigurasi gateway

Langkah pertama tentunya Anda harus melakukan konfigurasi gateway (*internet sharing*) sebagai berikut:

**Mengaktifkan IP Forward** sebagai berikut:

```
# echo 1 > /proc/sys/net/ipv4/ip_foward
```

atau

```
# sysctl -w net.ipv4.ip_forward=0
```

Mengaktifkan IP Masquerade sebagai berikut:

```
# iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
```

Sebagai catatan seluruh perintah diatas tidak bersifat permanen, untuk itu sebainya perintah perintah tersebut Anda tulis juga kedalam file /etc/rc.local

## ***Konfigurasi squid transparent proxy***

Langkah selanjutnya adalah mengatur squid agar bekerja dalam mode transparent, untuk itu Anda harus mengedit file /etc/squid/squid.conf, beberapa parameter konfigurasi squid yang perlu disesuaikan atau diatur ulang adalah sebagai berikut:

```
http_port 3128 transparent
```

```
acl mynetworks src 192.168.1.0/24
```

```
http_access allow mynetworks
```

Selanjutnya mendefinisikan rule firewall yang akan me-*redirect* trafik http ke proxy, untuk itu jalankan perintah seperti berikut ini:

```
# iptables -t nat -A PREROUTING -p tcp -s 192.168.1.0/24 -d ! 192.168.1.1 -  
dport 80 -j REDIRECT --to-ports 3128
```

## ***Mengatur database***

Langkah selanjutnya mengatur database dan membuat script rewrite.php serta script php yang terkait (aplikasi login).

- Membuat database 'capo' (diasumsikan user root mysql memiliki password 'kuci'):

```
# mysqladmin -u root -pkunci create capo
```

- membuat file capo.sql dengan isi file sebagai berikut:

```
CREATE TABLE `logon` (  
  `id` int(11) NOT NULL auto_increment,
```

```

`username` varchar(20) default NULL,
`ip` varchar(16) default NULL,
`logtime` datetime default NULL,
PRIMARY KEY (`id`)
) ENGINE=MyISAM AUTO_INCREMENT=485 DEFAULT CHARSET=latin1;

CREATE TABLE `user` (
  `id` int(11) NOT NULL auto_increment,
  `username` varchar(20) default NULL,
  `password` varchar(72) default NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `username` (`username`)
) ENGINE=MyISAM AUTO_INCREMENT=4 DEFAULT CHARSET=latin1;

INSERT INTO `user` VALUES (1,'henry','027e4180beedb29744413a7ea6b84a42'),
(2,'hana','0cc175b9c0f1b6a831c399e269772661');

```

- Membuat tabel user dan logon :

```
# mysql -u root -pkunci capo < capo.sql
```

## **Membuat script-script php**

Langkah selanjutnya adalah membuat beberapa script php yang diantaranya adalah rewrite.php, index.php, login.php dan logout.php. Isi dari file script php tersebut sebagai berikut:

- File rewrite.php adalah file utama yang akan memproses url request yang diterima dari squid melalui standard input. Url request yang diterima oleh rewrite.php berupa string single line yang memiliki format terdiri dari:
  - *Request-URI*
  - *Client IP address and fully qualified domain name*
  - *User's name, via either RFC 1413 ident or proxy authentication*
  - *HTTP request method*

Contoh url request :

***http://www.contoh.com/index.html 192.168.1.3/user.host.name henry GET***

Script rewrite.php harus menghasilkan outout berupa string single line yang menu njukkan

sebuah Request-URI dan diikuti sebuah baris baru seperti berikut ini:

**<http://www.contoh.com/index.html>**

Untuk itu buatlah script rewrite.php dengan isi file sebagai berikut:

```
#!/usr/bin/php
<?php
$con=mysql_connect("localhost","root","kunci");
$db=mysql_selectdb("capo");
$temp = array();
while ( $input = fgets(STDIN) ) {
    $temp = split(' ', $input);
    $ips = split('/', $temp[1]);
    $ip = $ips[0];
    $uri=$temp[0];
    $sql="SELECT count(*) FROM logon WHERE ip='$ip'";
    $q=mysql_query($sql);
    $data= mysql_fetch_row($q);
    if ($data[0][0] >= 1)
        {
            $output = $temp[0]."\n";
        }
    else {
        $output = "302:http://192.168.1.1/capo/index.php?
uri=$uri\n";
    }
    echo $output;
}
```

➤ File index.php, adalah script login form, dengan isi file sebagai berikut:

```
<html>
<head>
<title> Login form </title>
</head>
<body>
<table bgcolor='#ffffff' border='0' cellspacing='0' cellpadding='0'
width='100%'><tr><td align='center'><center><img src='logo.png' width='340'
height='111' /><br/>
<br>
<table width=45%><tr><td bgcolor='#dcdcdc' align='center'><font size=4
face=arial color=blue><b>Research And Development
```

```

Division<br></td></tr><tr></tr><tr> <td align='center'>
<div style='border:1px dashed red;margin-top:1em;background:lightyellow'>

<font color=blue><b>Captive Portal - Login<b></b>
</td>

    <table width= align='center' cellpadding=3 cellspacing=3 border=0>
    <form action=login.php method=POST>
    <input type="hidden" name="uri" value="<?php echo $_GET['uri'];?>">
<center>
    <tr>
        <td width=30% >Username</td>
        <td width=3>:</td>
        <td><input type=text name=username maxlength=100 size=10></td>
    </tr>

    <tr>
        <td width=30% >Password</td>
        <td width=3 align=center>:</td>
        <td><input type=password name=password maxlength=100 size=10></td>
    </tr>
    <tr>
        <td width=100% align=center colspan=3><input type=submit
class=button value=Login></td>
    </tr>

    </form>
    </table>
</body>
</html>

```

- File login.php, file ini akan memverifikasi user dan password yang disuplai pada form login. Isi file login.php seperti berikut ini:

```

<?
$con=mysql_connect("localhost","root","kunci");
$db=mysql_selectdb("capo");

$username = trim($_POST[username]);
$password = trim($_POST[password]);
$uri = trim($_POST[uri]);
session_start();
if ($_POST[username] && $_POST[password])
{
    $sql="select 1 from user where username='$username' and
password=md5('$password')";
    $q=mysql_query($sql);
    $data=mysql_fetch_row($q);
    if ($data[0][0]==1)
    {
        $_SESSION['ip']=$_SERVER["REMOTE_ADDR"];
    }
}

```

```

        $ip=$_SESSION['ip'];
        $sql="INSERT INTO logon
VALUES(0,'$username','$ip',now())";
        mysql_query($sql);
        header("location:$uri");

    }
    else
    {
        header("location:index.php?uri=$uri");
    }
}
else
{
    if (session_is_registered('ip'))
    {
        echo "Anda sudah berhasil login, dan dapat mengakses
internet dari komputer " .$_SESSION['ip'];
        echo "<BR> <B><i>Jangan ditutup halaman ini , silahkan
browsing</i></B>";
        echo "<br>Jika Anda mau logout klik di <A
HREF='logout.php'>sini</a>";

    }
    else
    {
        echo "Anda belum login, maka Anda tidak akan dapat
mengakses internet";
        echo "<br> jika Anda ingin mengakses internet silahkan
login dahulu, klik di <a href='index.php'>sini</a>";

    }
}
?>

```

- File logout.php, isi dari file logout.php adalah seperti berikut ini:

```

<?php
$con=mysql_connect("localhost","root","kunci");
$db=mysql_selectdb("capo");

session_start();
$ip=$_SERVER["REMOTE_ADDR"];
$sql="DELETE FROM logon WHERE ip='$ip'";
mysql_query($sql);
session_destroy();
echo "Anda telah berhasil logout maka akses internet berakhir, <br>
jika Anda ingin mengakses internet silahkan login dahulu, klik di <a
href='index.html'>sini</a>";
?>

```

File `rewrite.php` selanjutnya disalin kedalam direktori `/usr/lib/squid`, sedangkan file `index.php`, `login.php` dan `logout.php` disalin kedalam direktori penyimpanan dokumen web (`DocumentRoot`).

### ***Mendefinsikan url\_rewrite\_program***

Langkah terakhir adalah mendefinsikan url rewerite program pada squid dimana nama program rewirte nya adalah `rewrite.php`. Untuk itu Anda harus mengedit file `/etc/squid/squid.conf` dan aturlah parameter `url_rewrite_program` seperti berikut ini:

```
url_rewrite_program /usr/lib/squid/rewrite.php
```

```
url_rewrite_children 10
```

Setelah itu Anda dapat mengaktifkan squid seperti berikut ini:

```
# service squid start
```

atau

```
# /etc/init.d/squid start
```

dan untuk memungkinkan user login terlebih dahulu melalui halaman login yang telah dibuat maka service http harus diaktifkan juga, lakukan perintah berikut:

```
# service httpd start
```

atau

```
# /etc/init.d/httpd start
```

### ***Membuat jadwal tugas menghapus entri tabel logon***

Agar pada hari berikutnya klien harus login lagi untuk dapat mengakses web maka buatlah jadwal tugas yang akan selalu menghapus seluruh entri pada tabel logon setiap hari pada jam 01:05, untuk itu gunakan perintah 'crontab -e' kemudian buat jadwal tugas seperti berikut ini:

```
5 1 * * * mysql -u root -pkunci capo -e "delete from logon"
```

Selamat mencoba