No Cat Auth authenticates WLAN users against NAT routers

The Cat's Whiskers

Wireless Internet access is no longer a technical challenge, but wireless networks do face security problems. It is not a good idea to give free Internet access to the customers in the coffee shop across the road. An authentication method can prevent this – preferably a method that is platform independent.

BY JOCHEN STÄRK

It is common knowledge that WLAN encryption and authentication techniques have as many holes as a Swiss cheese [1] and can be cracked in next to no time [2]. Tools such as Airsnort, D-Wepcrack, and Kismet [3] provide ample evidence of this. WLANs must therefore be regarded as external, insecure networks. If you intend to use your WLAN to access your internal network from your laptop, it makes sense to adopt VPN technologies [4] such as IPsec or OpenVPN.

However, if you need a WLAN hotspot to provide Internet access to multiple users, a VPN is typically unnecessary. Users can and should choose a suitable security technology themselves. But this does not remove the need for authentication – after all, you will not want to provide free Internet access to anyone in the vicinity. No Cat Auth [5] is the answer. Clients wanting to use this tool need only a Web browser.

No Cat Auth can authenticate users against MySQL databases, shadow passwords, LDAP, IMAP, PAM, Samba, or NIS (see Table 1) and restricts the amount of bandwidth available to a user. The software consists of a gateway and an authentication server. The gateway is an adaptive firewall that controls the interface between the WLAN and the Internet, allowing only packets from users who have authenticated via a Web interface to pass.

If the No Cat Auth server uses a MySQL or PostgreSQL table as its user database, users who need access are first required to fill out a registration form. Admins can assign additional rights to accounts created this way at a later stage. *admintool*, and a few external tools such as PHP-My-Admin are useful in this case. *admintool* helps you manage your No Cat accounts (see Table 1). If a user directory already exists (LDAP,

Table 1: Authentication

Feature	Description	Status	Admin tools
Radius	Authentication via a Radius server	experimental	n.a.
LDAP	Authentication via LDAP	n.a.	No
DBI::MySQL	User database in MySQL with full support for authentication and registration	Full support	n.a.
DBI::PostgreSQL	User database in PostgreSQL with full support for authentication and registration	Full support	n.a.
Passwd	Authentication via local files	Full support	Full support
PAM	Authentication via PAM	n.a.	n.a.
Samba	Authentication via Samba or Windows (Primary Domain Controller)	n.a.	No
IMAP	Authentication via IMAP accounts	n.a.	No
NIS	Authentication via NIS	Full support	No

THE AUTHOR

Jochen Stärk works as a sales engineer for Borland GmbH in Germany, where his major focus is C++ Builder and Kylix.

PAM, a Windows PDC, Imap, Radius or NIS), you can tell the authentication server to access that directory.

When a user on a No Cat network uses her browser to access a website, the IPTables-based No Cat gateway forwards the request to the authentication server. The user is prompted to enter her ID and password. If the ID and password check out, the user (or her computer to be more precise) is allowed onto the Internet via the gateway. The gateway daemon enables NAT (Network Address Translation), and permits use of common protocols such as SSH, POP3, IMAP, HTTP, HTTPS, passive FTP, and many more. SMTP is disabled by default to provide spam protection.

Easy Webmin-based Configuration

A Webmin module [9] removes the need to set up the No Cat infrastructure manually. Despite the relatively low version number (0.51), it is quite stable and available for immediate use. Simply modify the paths to your configuration files. The following example assumes a DHCP server with an address scope of 192.168.0.10 through 192.168.0.254. The user database is hosted on a server with an internal IP address (192.168.0.1). The server also runs MySQL, PHP-MyAdmin, and Apache with the Mod_SSL module.

Let's not set too hard a target for our first exercise; all we need is an open gateway. The gateway is easy to install, but watch out for the pitfalls – No Cat will install both the gateway and the authentication server in */usr/local/nocat*. The packages are distinguished by a prefix:

```
tar xvfz NoCatAuth-0.82.tar.gz
cd NoCatAuth-0.82
make PREFIX=/usr/local/nocat/gw
 gateway
```

You can enable the open gateway by adding a *GatewayMode Open* entry to the */usr/local/nocat/gw/nocat.conf* configuration file. No Cat supports three modes:

- Open: The gateway displays a welcome page, and prompts the user to accept the conditions of use.
- Passive: Requires users to authenticate. This is the recommended setting.
- Captive: Like Passive, but does not support NAT.

Now type */usr/local/nocat/gw/bin/gateway* to launch the gateway.

Box 1: Project Name

No Cat is the name of a community that runs hotspots in Sonoma County, CA. Its members provide mutual free wireless Internet access.

A quick glance at the project homepage at *http://nocat.net/* explains the unusual name. It originates from a famous quotation. When asked how radio works, Albert Einstein replied: "You see, wire telegraph is a kind of a very, very long cat. You pull his tail in New York and his head is meowing in Los Angeles. Do you understand this? And radio operates exactly the same way: you send signals here, they receive them there. The only difference is that there is no cat."

User Database

As the gateway will not typically be required to provide access to any laptop in the vicinity, it makes sense to create a user database. In *GatewayMode Passive* mode, the gateway will access the authentication server and ask the server to check out the user. *AuthServiceAddr 192.168.0.1* tells the gateway where to look for the server.

In a larger community it is not desirable to have each hotspot provider run their own authentication server. A central server means that members need only one account, and single sign-on makes life easier for the community. To run an authentication server, the first step is to compile the server:

```
make PREFIX=/usr/local/nocat/as
 authserv
make PREFIX=/usr/local/nocat/as
 pgpkey
chown -R wwwrun /usr/local
/nocat/as/pgp
cp /usr/local/nocat/as/
trustedkeys.gpg /usr/local/
nocat/gw/pgp
```

To allow only authenticated users, rather than a public access point, add a line with *MembersOnly 1* to your configuration in */usr/local/nocat/gw/nocat.conf*. A few changes to the authentication server are also needed: the Apache server needs to be able to run the CGI scripts and serve up the correct HTML files. */usr/local/nocat/as/httpd.conf* provides a template for these changes.

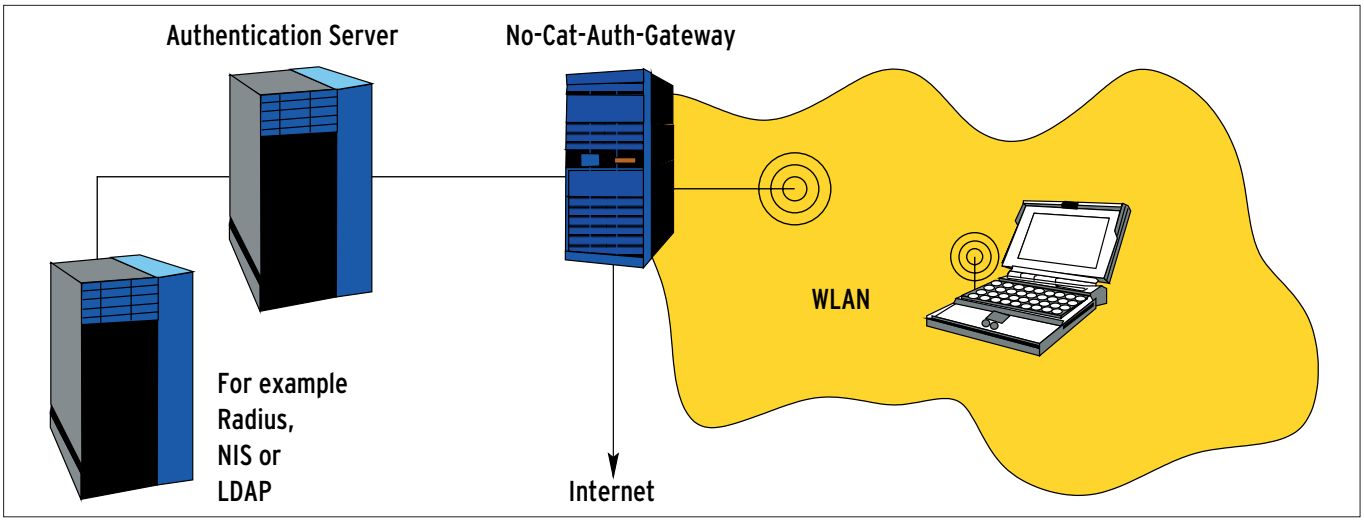

Figure 1: Wireless clients use the access point to access the Internet. Before doing so, they first have to negotiate the No Cat authentication gateway. The authentication server decides if they are allowed to do this.

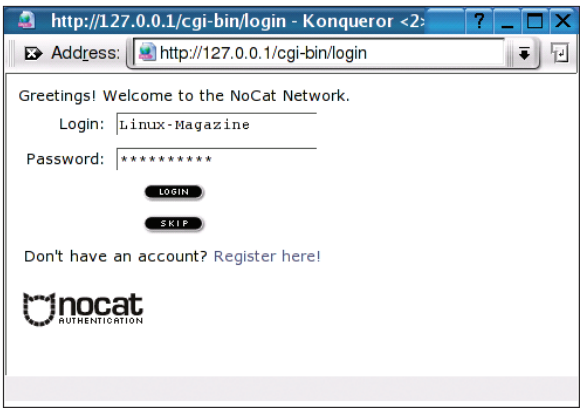

Figure 2: Visitors to the No Cat network are first shown the welcome screen. Users can sign on and then access the Web via the No Cat gateway. Admins can modify the HTML sources and error messages to reflect their own requirements.

The user database also needs some attention. Use PHP-My-Admin for this task. The MySQL front-end allows admins to create a database user and a database for No Cat, then creates the required tables (see the MySQL dump in the source package below *etc/nocat.schema*), and modifies the authentication gateway configuration (*/usr/local/nocat/as/nocat.conf*):

```
DataSource DBI
Database dbi:mysql:database
=nocat
DB_User nocat
DB_Passwd strokethecat
```

You can now run the Apache server as your authentication server (*apachectl startssl*). The gateway initially forwards user requests to the Apache server (see Figure 1) where users are prompted to sign on (see Figure 2) or at least accept the conditions of use.

The No Cat homepage at [5], various mailing lists and Toni Diaz's [6] Howto provide more details. Strangely, the initiator of No Cat, Rob Flickenger, dedicates a mere three pages of his own book [10] to No Cat Auth.

Wired Networks

No Cat is designed for access points configured as bridges. This means the software is also useful as a gateway for other network technologies.

No Cat's design allows it to provide central authentication facilities to geographically distributed user groups. It distinguishes between logged on and not logged on users, and can provide Quality of Service support. Traffic control (TC) allows admins to restrict the data rate per group (total, owner, and public), although this feature is currently experimental.

A patch [7] provides add-on accounting support, allowing admins to log data transfer volumes. Accounting is independent of the authentication method used, and will log to Radius or to files (*File*). Unfortunately, we were unable to confirm whether file-based accounting works. It seems that accounting to the No Cat log (*Log*) and to databases (*DBI*) is planned, but not yet implemented.

When a user signs on or off, the No Cat gateway logs this event in the *nocat.log* file. The name and path are configurable using the *GatewayLog* option, and there is a *Verbosity* switch. You can also use *LogFacility* to tell No Cat to log to the Syslog.

This does make it difficult to provide central login statistics, and was what prompted me to develop a patch [8] that logs the authentication time, MAC and IP address, and login refresh time on a file per-user basis. A second file reports the current status for each user (logged on, or not logged on).

Admins can then use a statistics page (an example is available from [8]) to discover who is online, if a user is allowing other users to share her account (as evidenced by different MAC addresses), and how long each user was logged on to the hotspot.

Box 2: Security

No Cat Auth security cannot be compared with a good client-to-site VPN. This is due to the fact that No Cat does not provide a cryptographic link between authentication and data transfer. The authentication itself is SSL protected (assuming a genuine certificate). But No Cat does not provide data transfer control, and above all does not support encryption. Any data transferred by hotspot users can be sniffed, and additional encryption facilities are strongly recommended.

After authenticating a user, the No Cat gateway enables the user's IP and MAC addresses. But an attacker would be able to sniff these data and spoof both the IP and the MAC address gaining access to the internal network behind the gateway until the regular user's authentication timed out. This is an acceptable risk for most WLAN communities. If you need more security, you should look to IPsec, OpenVPN, another secure VPN technology [4].

Conclusion

No Cat Auth is useful for networks where central access controls can and should be applied. Clients need only a browser. But there is a downside: the lack of a client that can sign on automatically or prevent a timeout. Users need to have a browser window open to keep the connection alive.

Meanwhile, development on a new implementation is in progress. While No Cat Auth was written in Perl, No Cat Splash uses ANSI C and multithreading, and will be especially suited to embedded applications. However, the authors will continue to support No Cat Auth. ■

INFO

[1] Scott Fluhrer, Itsik Mantin, and Adi Shamir, "Weaknesses in the Key Scheduling Algorithm of RC4": *http://www.crypto.com/papers/others/rc4_ksaproc.ps*

[2] Adam Stubblefield, John Ioannidis, and Aviel D. Rubin, "Using the Fluhrer, Mantin, and Shamir Attack to Break WEP": *http://www.cs.rice.edu/~astubble/wep/*

[3] Mark Vogelsberger, "Kismet and Co: Focus on WLAN Security", Linux Magazine issue 38, p. 40

[4] VPN technologies, "How secure is your VPN software really?", Linux Magazine issue 39, p.48.

[5] No Cat Auth: *http://nocat.net/wiki/*

[6] Howto for No Cat Auth: *http://blyx.com/public/wireless/nocatbox/nocatbox-howto-en.pdf*

[7] Patches for IPFW2, Accounting, Radius, and SNMP-Monitoring: *http://www.pogozone.net/projects/nocat/*

[8] Patch for logging user access: *http://www.usegroup.de/software/nocat/*

[9] Webmin module for No Cat: *http://sourceforge.net/projects/nocat-webmin/*

[10] Rob Flickenger, "Building Wireless Community Networks. Planning and Deploying Wireless Local Networks", O'Reilly 2003, chap. 7, p. 113ff