

Re: remote port unlocker – does such a thing exist??

Source: <http://www.derkeiler.com/Newsgroups/comp.security.firewalls/2002-10/4775.html>

From: paul@nowhere.invalid

Date: 07/15/02

From: paul@nowhere.invalid

Date: Mon, 15 Jul 2002 10:38:44 +0000 (UTC)

In article <1e41930a.0207101247.17e1fba1@posting.google.com>, Ant wrote:

> *Thanks for taking the time to post that very useful reply Scott.*

>

> *Disappointing to learn that there isn't a utility on Windows that*

> *allows me to do that.*

There is a utility that does *exactly* what you want, but it only runs on linux at the moment. It doesn't generate a response to any traffic, and is completely invisible unless you know how to trigger it, a one time password (prevents replay attacks) and a public/private key pair (prevents anyone knowing what you're doing). Here's the README:

```
# $Id: README,v 1.2 2001/07/13 18:22:48 paul Exp $
```

HolePunch is a simple daemon that executes an operation after three authentication processes are performed on an arbitrary packet. It is the responsibility of some other system to deliver selected packets to the daemon from which, providing the packet is large enough, the end of the packet will be sent to the authentication system. Therefore, ICMP, TCP, UDP or any other protocol can be used to deliver the request.

The program can gather input packets using the iptables QUEUE module, via stdin, or via libpcap.

i.e.

```
iptables -A INPUT -p icmp -j QUEUE
```

After selecting the appropriate packets, which, if iptables is used, could also involve a rate limiting mechanism, the authentication processes are applied:

1) The "trigger" process is intended to be a rapid, simplistic mechanism to ensure that processor intensive operations are not performed needlessly. Currently, it merely checks the IP ID, but perhaps something such as a shared secret depending on the date and time could be used.

comp.security.firewalls: Re: remote port unlocker – does such a thing exist??

2) To prevent interception of the request and thus knowledge that could aid a later spoofing attack, the request is encrypted using the RSA public key of the server.

3) After decrypting the request using the server's private key, the decrypted data is copied to a command structure. This structure must contain a valid OPIE one time password corresponding to the transmitted userid.

Requirements:

OPIE

The install process for OPIE updates login and su, and does not install opie.h nor libopie.a. It may be preferable to do the installation manually including:

```
cp opie.h /usr/local/include
cp libopie/libopie.a /usr/local/lib
umask 077
mkdir /etc/opielocks
touch /etc/opiekeys
```

iptables – if using iptables

Installation of iptables should install the include files and library modules in the appropriate place (needed on server only).

GMP

The GNU multiple precision arithmetic library is required to perform RSA.

Libnet

Libnet is used to generate the client request (needed on clients only).

Risk analysis:

The intention of this program is to enable limited services to be opened as and when required instead of being enabled full time. This would allow numerous VPNs and other services such as SSH or DNS zone transfer to be installed even though the services themselves may contain as yet unknown flaws and may need to be accessed from IP addresses that are not fixed and can be spoofed.

HolePunch should prevent flaws that cannot be easily audited in many diverse programs, providing of course, that HolePunch itself is not vulnerable. HolePunch presents several threats that need to be examined:

1) The trigger may be determined and a denial of service attack perpetrated.

Re: remote port unlocker – does such a thing exist??

comp.security.firewalls: Re: remote port unlocker – does such a thing exist??

This threat would require either network sniffing, or compromise of either the client or server. Given knowledge of the trigger, it could be possible to exhaust the CPU resources of the server in order to verify malicious requests.

This can be mitigated by making the trigger event harder to predict, and/or by rate limiting the acceptance of a packet to be checked.

- 2) It may be possible to generate a buffer overflow in the server by sending a packet that, after satisfying the trigger, is decrypted into an undersize buffer and perhaps copied into other structures.

Checks in decrypt() are intended to prevent this.

- 3) Flaws in the gmp library could conceivably generate an unknown exception and execute arbitrary code.

The client does not know the server's private exponent, and thus is unlikely to be able to predict the result of a gmp operation. If the client already knew the servers private key, this avenue may be more likely however, in this case, it is more likely that the server is already compromised.

- 4) Bugs in HolePunch could include buffer overflows in the packet copying routines, and:

- 5) Bugs in HolePunch could include buffer overflows and logic flaws that would allow an authenticated user to perform operations that should not be allowed (i.e. executing as root instead of their uid).

Attempts have been made to ensure that data copies are correctly constrained and that credentials are correctly set prior to executing as the authenticated user. Apart from SYSLOG, all logic and policy is intended to be implemented by external programs running with the credentials of the authenticated user.

HolePunch implements four commands:

OPENPORT – Open a port from a specified IP for a specified duration
SYSLOG – Log a message to syslog
RELAY – Relay a command to another machine
RUN – Run a program

SYSLOG is implemented internally, OPENPORT, RELAY and RUN require external programs to provide the required functionality as well as perhaps specifying additional rules and restrictions. Prior to executing external programs, HolePunch will set user credentials according to the authenticated uid and OTP. If these programs require root access they must be setuid root.

Each program will be run with no arguments and parameters set as environment variables, as follows:

Re: remote port unlocker – does such a thing exist??

comp.security.firewalls: Re: remote port unlocker – does such a thing exist??

HP_UID – UID specified by caller and corresponding to OTP

HP_PROTO – IP Protocol as an integer. See /etc/protocols.

HP_IPADDR – IP address specified by caller, as an integer.

HP_PORT – Port specified by caller. Can be used as a protocol specifier
or ICMP type.

HP_VALIDITY – Validity of command in seconds.

HP_CMD – Command number.

HP_RELAY_CMD – if HP_CMD is CMD_RELAY, HP_RELAY_CMD is intended command

HP_RELAYIP – if HP_CMD is CMD_RELAY, HP_RELAYIP is intended destination

HP_MSG – Short message or command to execute.

The intention of the RELAY command is for the use by a machine that can indelibly log a command by an authenticated source and then forward a new request, authenticated by the relay, to the intended destination.

i.e. the relay machine may allow user fred to shutdown a machine, but only if it has a permanent log of the request first.

Install the prerequisite programs, configure /etc/holepunch with an RSA private key of 1024 bits then create an opie passwd for any desired user with opiepasswd -c.

Generating an RSA public/private key pair:

Any program can be used to do this, however FreeS/Wan includes a program:

rsasigkey that generates the appropriate values, where

Modulus = n

PublicExponent = e

PrivateExponent = d

Client:

holesend creates a packet to send to the HolePunch daemon, inserts an OPIE one time password and then encrypts it with the public key of the server

Usage:

```
./holesend -h host [-k keyfile] -c run|log|open|relay [-m message|command] \  
[-o opie_passphrase] [-u uid] [-C run|log|open] [-v validity] \  
[-n protonum|IP|TCP|ICMP] [-p port] [-i ipaddr|hostname] \  
[-I relayipaddr|hostname] [-t trigger] \  
[-s seqno] [-S seed] [-P OPIE_key] [-e exponent] [-N modulus]
```

keyfile consists of the following lines

host.e = public exponent

host.n = public modulus

host.trigger = default trigger

host.validity = default validity time

host.seed.uid = opie seed

if keyfile is not specified, and e, N and seed or opie key, are not all specified, holesend will attempt to read ~/.holesend

Re: remote port unlocker – does such a thing exist??

comp.security.firewalls: Re: remote port unlocker – does such a thing exist??

Examples:

Assuming ~/.holesend contains the default parameters for a host example.com

```
./holesend -h example.com -c open -n tcp -p 22 -i me.myisp.com -s 499
```

This will send a request to example.com to open tcp/22 from me.myisp.com
499 is the opie sequence number to use and the passphrase will be requested.

```
./holesend -h logger.example.com -c relay -C log -I foo.example.com \  
-m "hello world" -s 498
```

This will send a request to logger.example.com asking it to relay a request
to foo.example.com to write a message to syslog. If logger agrees, it
will generate a new request and send it to foo.example.com. foo.example.com
would typically not allow a request from the original host/user.

TODO:

Autoconf and port

Installation:

Configure Makefile with -DUSE_IPQ or -DUSE_PCAP
make

copy holesend and holepunch to a suitable binary directory

generate rsa key

configure /etc/holepunch

copy/create/configure example helper scripts (see ./examples/)

start up

Note: /etc/holepunch should not be readable by anyone other than root

- *Next message:* Joe: "Re: Limit Access?"
- *Previous message:* chuck: "Re: Plug & Play"
- *In reply to:* :"Re: remote port unlocker – does such a thing exist??"
- *Next in thread:* Ant: "Re: remote port unlocker – does such a thing exist??"
- *Messages sorted by:* [date] [thread] [subject] [author] [attachment]