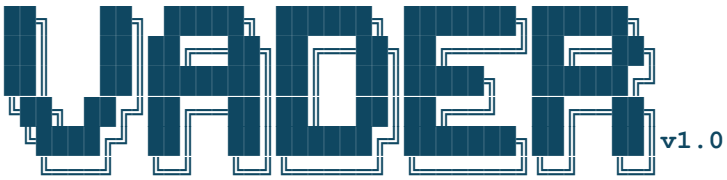


Advanced Command Builder Tool for NMAP

Ahmad Kamal

CONTENTS

1. Introduction 5
   Key Features 5
   Target Audience 5
2. Installation and Requirements 6
   System Requirements 6
   Installation Steps 6
   First Run 6
3. Getting Started 7
   Basic Workflow 7
   Interactive Interface 7
   Command Line Arguments 8
4. Command Reference 8
   1. Target Management Commands 8
   2. Scan Type Commands 9
   3. Port Specification Commands 10
   4. Script & Enumeration Commands 11
   5. Stealth & Evasion Commands 12
   6. Host Discovery Commands 13
   7. Output & Reporting Commands 13
   8. Utility Commands 14
   9. Profile Commands 15
   10. Command Management 16
   11. Help & Information 17
   12. Validation 17
5. Detection Level System 18
   Overview 18
   Detection Levels Explained 18
   Detection Level Modifiers 19
   Practical Use 19
6. Scan Types & Techniques 20
   TCP Scanning Methods: 20
   UDP Scanning 21
   Host Discovery Methods: 21
   Advanced Scanning: 21

7. STEALTH AND EVASION 22
Timing Templates 22
Packet Manipulation Techniques 23
IP Spoofing & Decoys 23
Timing & Rate Control 24
Combining Techniques for Maximum Stealth 24
Evasion Strategy by Environment 24
8. SERVICE ENUMERATION 25
Web Application Scanning 25
SSL/TLS Assessment 25
Windows/SMB Enumeration 25
SSH Enumeration 26
FTP Enumeration 26
DNS Enumeration 26
Database Enumeration 27
Email Service Enumeration 27
Custom Script Usage 27
9. OUTPUT AND REPORTING 28
Output Formats 28
Report Generation 29
Verbose Output Control 29
Output Management Best Practices 30
10. QUICK SCAN PROFILES 31
Available Profiles 31
Profile Customization 31
Profile Selection Guidelines 32
VADER – Practical Examples Quick Reference 33
12. BEST PRACTICES 35
Pre-Scan Preparation 35
Legal and Authorization 35
Technical Preparation 35
Documentation Requirements 35
Scan Execution Guidelines 35
Timing and Scheduling 35
Detection Management 35

Network Considerations 36
Results Analysis 36
Data Validation 36
Security Assessment 36
Documentation Standards 36
Reporting and Communication 36
Report Structure 36
Communication Protocols 36
Stakeholder Management 37
Tool Management and Maintenance 37
Configuration Management 37
Security Considerations 37
Performance Optimization 37
13. TROUBLESHOOTING 38
Common Issues and Solutions 38
Performance Optimization 40
Error Messages and Solutions 40
Debug Mode 41
Log Analysis 41
Getting Help and Support 41
14. LEGAL AND ETHICAL CONSIDERATIONS 42
1. Legal Framework 42
2. Compliance Considerations 42
3. Legal Risk Management 43
4. Ethical Guidelines 43
5. Minimizing Impact 43
6. Best Practices for Ethical Scanning 44
7. Industry Standards and Frameworks 44

1. INTRODUCTION

VADER (Version 1.0) is an advanced, interactive command-line tool designed to simplify the creation and execution of **Nmap** commands for ethical hackers, penetration testers, and security professionals.

Unlike traditional Nmap usage that requires memorizing complex syntax and flags, VADER provides an intuitive, menu-driven interface with built-in detection level awareness and comprehensive guidance.

KEY FEATURES

- **Interactive Command Building** – Menu-driven interface eliminates the need to memorize Nmap syntax.
- **Detection Level Awareness** – Real-time feedback on how detectable your scans are (0–5 scale).
- **Comprehensive Stealth Options** – Advanced evasion techniques to bypass firewalls and IDS.
- **Service-Specific Enumeration** – Targeted scripts for web, SMB, SSH, FTP, and DNS services.
- **Quick Scan Profiles** – Pre-configured scan templates for common scenarios.
- **Educational Components** – Built-in explanations and examples for learning.
- **Simple Report Generation** – Convert scan results into readable reports.

TARGET AUDIENCE

- Ethical hackers and penetration testers.
- Security professionals conducting authorized assessments.
- Network administrators performing security audits.
- Students learning network security and reconnaissance.
- Cybersecurity researchers and professionals.

2. INSTALLATION AND REQUIREMENTS

SYSTEM REQUIREMENTS

- **Operating System:** Linux, macOS, or Windows (with WSL).
- **Dependencies:** Bash shell (version 4.0 or higher), Nmap (latest version recommended).
- **Permissions:** Root privileges recommended for full functionality.
- **Storage:** Minimal disk space (< 1MB).

INSTALLATION STEPS

Step 1 – Install Nmap

```
# Ubuntu/Debian
sudo apt update && sudo apt install nmap

# CentOS/RHEL
sudo yum install nmap

# macOS
brew install nmap
```

Step 2 – Download VADER

```
# clone repository
git clone https://github.com/ahmadVader0/vader.git
```

Step 3 – Set Permissions

```
chmod +x vader.sh
```

Step 4 – Verify Installation

```
./vader.sh --version
```

FIRST RUN

```
# With root privileges
sudo ./vader.sh
# With command line options
./vader.sh --target 192.168.1.1 –quick
```

3. GETTING STARTED

BASIC WORKFLOW

1. **Set Target** – Specify the IP, hostname, or network to scan.
2. **Choose Scan Type** – Select appropriate scanning technique.
3. **Configure Ports** – Define which ports to scan.
4. **Add Scripts** – Include NSE scripts for enumeration.
5. **Apply Stealth** – Configure evasion techniques if needed.
6. **Set Output** – Define how results should be saved.
7. **Build & Execute** – Review and run the final command.

INTERACTIVE INTERFACE

COMMAND LINE ARGUMENTS

Argument	Description	Example
`--help`	Show help information	`Help`
`--scan`	Select scan type	`Scan --`
`--target`	Set target directly	`target 192.168.1.1`
`--profile`	Load quick scan profile	`Profile --`
`--stealth`	Select stealth option	`Stealth --`
`--execute`	Execute the command	`execute`

4. COMMAND REFERENCE

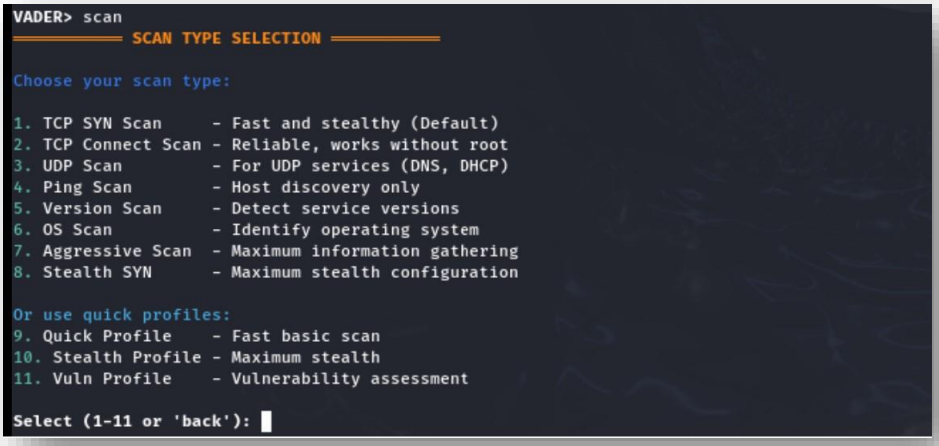
1. TARGET MANAGEMENT COMMANDS

Command	Purpose	Usage Example(s)	Supported Formats

`target`	Set the scan target(s)	`target 192.168.1.1`
		`target google.com`
		`target 10.0.0.0/24`
		`target file my_targets.txt`
			Single IP, Hostname, CIDR, IP range, File input

`exclude`	Exclude specific hosts from scanning	`exclude 192.168.1.1,192.168.1.5`	List of hosts/IPs

2. SCAN TYPE COMMANDS



Command: `scan` – Select scanning technique.
Usage: `scan <type>` or `scan` (interactive)

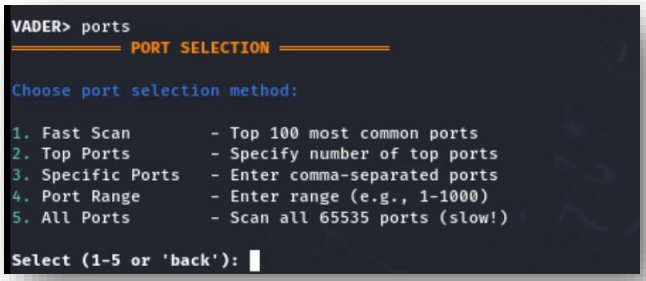
Available Types:

- **syn** – TCP SYN scan (DEFAULT, STEALTH)
- **connect** – TCP connect scan (RELIABLE)
- **udp** – UDP scan (SLOWER, FOR UDP SERVICES)
- **ping** – Host discovery only
- **version** – Service version detection
- **os** – Operating system detection
- **aggressive** – All features enabled

Examples:

```
VADER> scan syn
VADER> scan version
VADER> scan aggressive
```

3. PORT SPECIFICATION COMMANDS

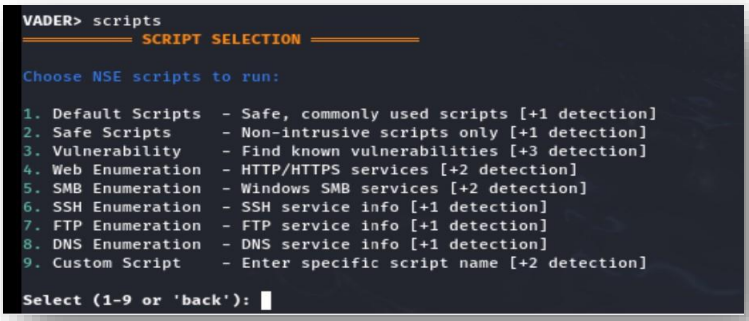


Command	Purpose	Usage Examples	Notes
ports	Define which ports to scan	ports 80,443,22,21 ports 1-1000 ports fast ports all	Supports single ports, ranges, and keywords

Port Keywords:

- **fast** – Top 100 ports
- **all** – All 65535 ports
- **top100**, **top1000** – Commonly used ports

4. SCRIPT & ENUMERATION COMMANDS



Command: `scripts` – Add NSE (Nmap Scripting Engine) scripts
Usage: `scripts <type>` or interactive

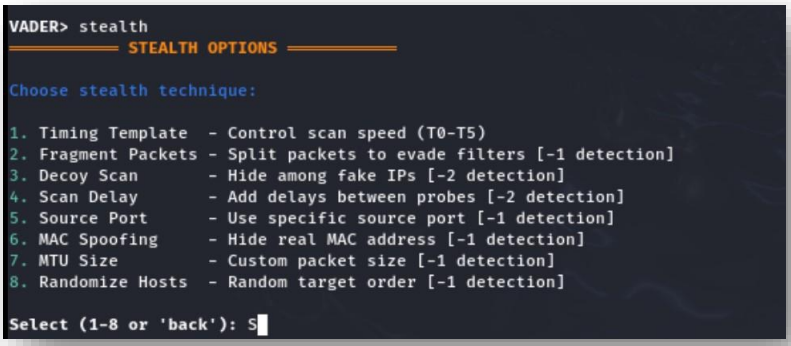
Script Types:

- `default` – Safe, common scripts
- `safe` – Non-intrusive only
- `vuln` – Vulnerability detection
- `web` – HTTP/HTTPS enumeration
- `smb` – Windows SMB enumeration
- `ssh` – SSH service enumeration
- `ftp` – FTP enumeration
- `dns` – DNS enumeration
- **Custom** – `scripts <script-name>`

Examples:

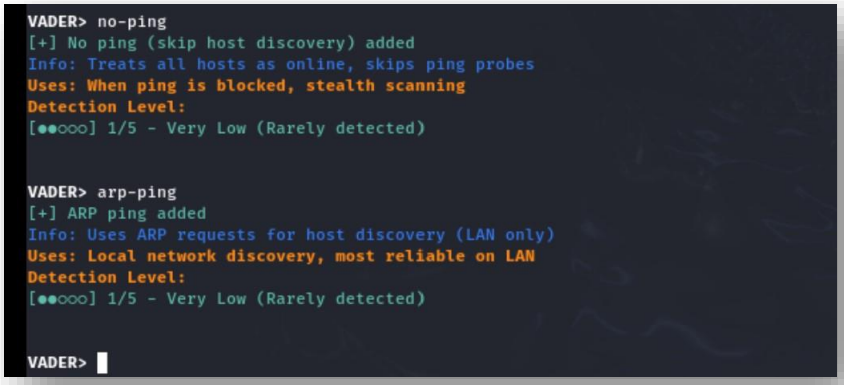
```
VADER> scripts vuln
VADER> scripts web
VADER> scripts http-title
```

5. STEALTH & EVASION COMMANDS



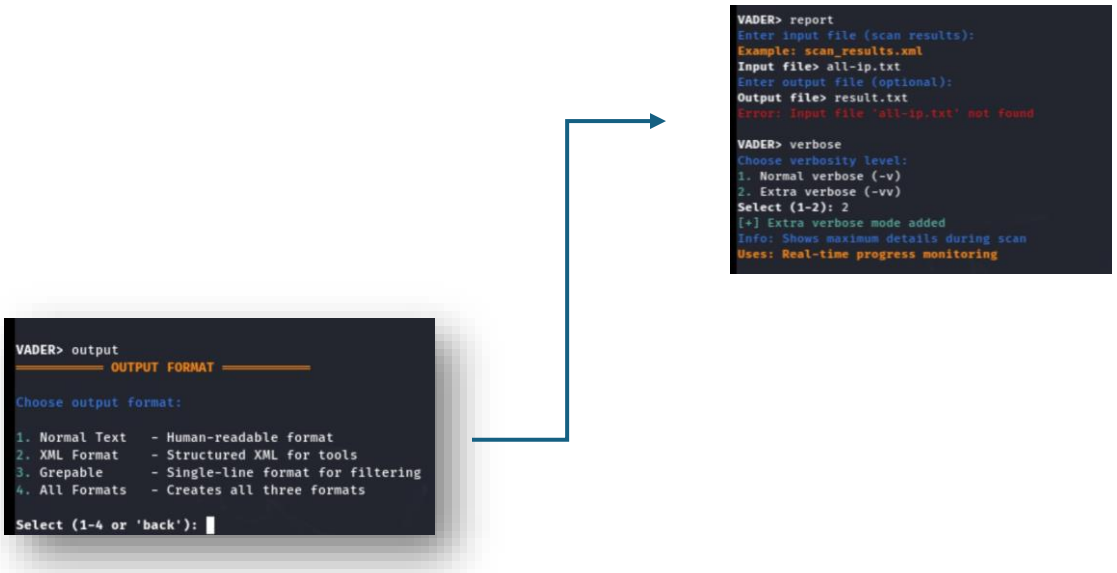
Technique	Purpose	Example Usage
`timing <0-5>`	Set timing template (0 = slowest, 5 = fastest)	`stealth timing 1`
`fragment`	Fragment packets to evade filters	`stealth fragment`
`decoy <ips>`	Use decoy IP addresses	`stealth decoy 1.2.3.4,5.6.7.8,ME`
`delay <time>`	Delay between probes	`stealth delay 10s`
`source-port`	Set custom source port	`stealth source-port 53`
`spoof-mac`	Spoof MAC address	`stealth spoof-mac`
`mtu <size>`	Set custom MTU size	`stealth mtu 24`
`randomize`	Randomize host scanning order	`stealth randomize`

6. HOST DISCOVERY COMMANDS



- **`no-ping`** – Skip host discovery (reduces detection by 1 level)
- **`arp-ping`** – ARP requests for LAN discovery (normal traffic)

7. OUTPUT & REPORTING COMMANDS



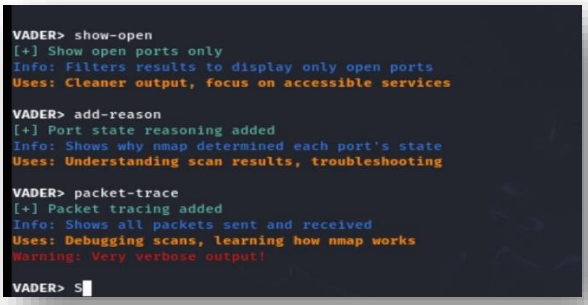
Command	Purpose	Examples

output	Configure output format & file	`output normal results.txt`
		`output xml scan_results.xml`
		`output all full_scan`

verbose	Enable verbose output	`verbose` or `verbose 2`
report	Generate reports from results	`report scan_results.xml report.txt`

8. UTILITY COMMANDS

- **show-open** – Display only open ports
- **add-reason** – Show reason for port state
- **packet-trace** – Debug by showing all packets (VERY VERBOSE)



9. PROFILE COMMANDS

- **quick** – SYN + Fast ports
- **stealth** – Max stealth config
- **full** – Comprehensive assessment
- **vuln** – Vulnerability assessment
- **discovery** – Network discovery only

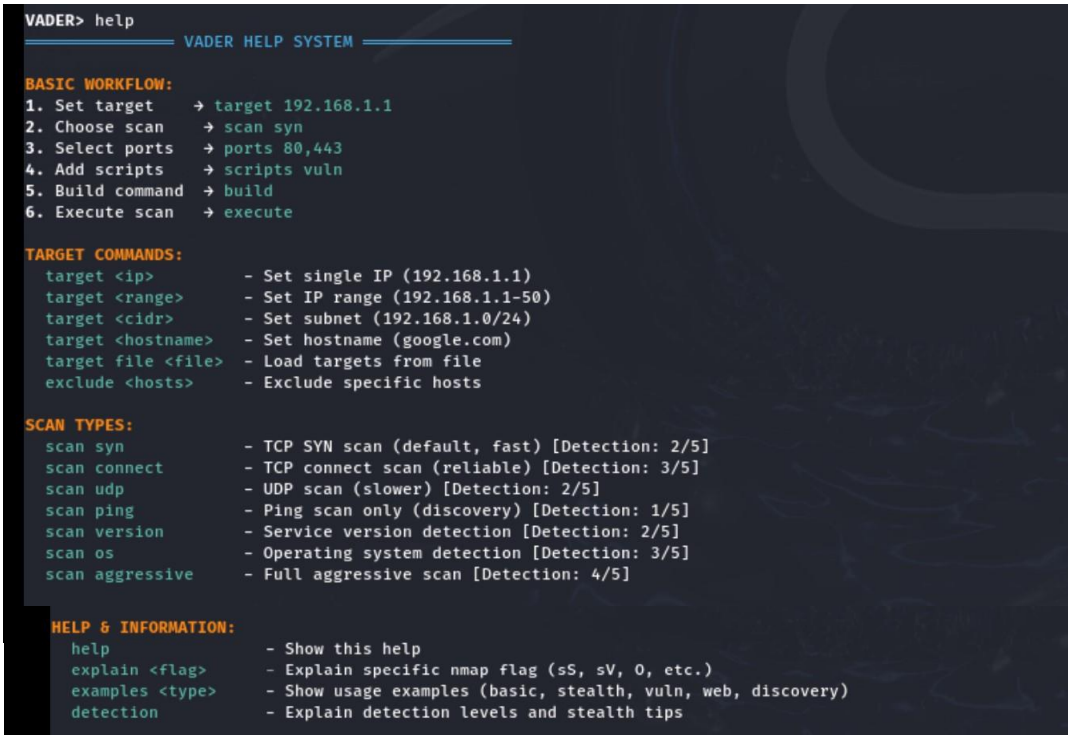
Usage:

```
profile quick
```

10. COMMAND MANAGEMENT

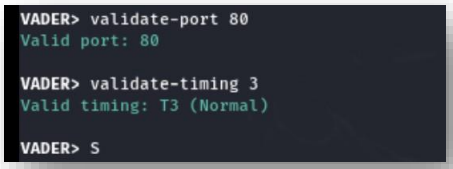
- **`build`** – Show full Nmap command without executing
- **`execute`** – Run built command (MAY CONFIRM HIGH-DETECTION SCANS)
- **`save <file>`** – Save command to file
- **`reset`** – Clear all settings
- **`status`** – Show current configuration

11. HELP & INFORMATION



- **`help`** – Full help menu
- **`explain <flag>`** – Explain an Nmap flag
- **`examples <type>`** – Predefined scan examples
- **`detection`** – Explain detection levels and stealth tips

12. VALIDATION

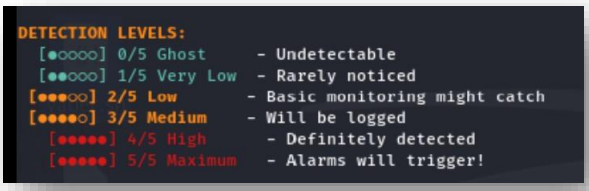


- **`validate-port <port>`** – Check port format
- **`validate-timing <0-5>`** – Check timing template value

5. DETECTION LEVEL SYSTEM

OVERVIEW

VADER implements a **0–5 Detection Level System** to help users balance **thoroughness vs. stealth** in scans.



DETECTION LEVELS EXPLAINED

Level	Name	Visibility	Description	Typical Techniques	Example Command
0	Ghost Mode	Extremely hard to detect	Uses advanced evasion techniques	Idle scans, T0 timing, max evasion	`stealth timing 0 + stealth fragment + stealth delay 60s`
1	Very Low	Rarely triggers alerts	Appears as normal traffic	Slow timing, basic enumeration	`Ping scans, SSH enumeration with T1 timing`
2	Low	Might be noticed by basic tools	Still fairly stealthy	Standard SYN scans, version detection	`scan syn + scan version`
3	Medium	Will likely be logged	Security teams may investigate	Connect scans, OS detection	`scan connect + scan os`
4	High	Definitely detected	May trigger immediate alerts	Aggressive scans, vuln scripts	`scan aggressive + scripts vuln`
5	Maximum	Guaranteed to trigger alarms	Security response expected	T5 timing, all ports, exploit scripts	`stealth timing 5 + ports all + scripts vuln`

DETECTION LEVEL MODIFIERS

REDUCE DETECTION (−1 TO −3)

- Slow timing templates (**T0, T1**): −1 to −2
- Packet fragmentation: −1
- Scan delays: −1 to −2
- Decoy scans: −2
- Source port spoofing: −1
- MAC address spoofing: −1
- Host randomization: −1
- Skip ping discovery: −1

INCREASE DETECTION (+1 TO +4)

- Fast timing (**T4, T5**): +2 to +3
- All ports scanning: +2
- Vulnerability scripts: +3
- Aggressive scanning: +4
- Multiple script categories: +1 each
- OS detection: +2
- Service version detection: +1

PRACTICAL USE

For Maximum Stealth

1. Start with levels 0–1
2. Use **T0/T1**
3. Stack multiple evasion methods
4. Limit port ranges
5. Use **safe scripts only**

For Comprehensive Assessment

1. Accept levels 3–4
2. Include version & OS detection
3. Add vuln scripts
4. Scan large port ranges
5. Generate detailed reports

For Balanced Approach

1. Aim for levels 2–3
2. Use **T2/T3**
3. Selective evasion
4. Focus on key services
5. Monitor for defense triggers

6. SCAN TYPES & TECHNIQUES

TCP SCANNING METHODS:

Method	Detection	Speed	Reliability	Requirements	Best For	Command
SYN Scan (-sS)	2/5	Fast	High	Root privileges	General purpose scanning (default)	`scan syn`
Connect Scan (-sT)	3/5	Medium	Very high	None	Non-root users, through proxies	`scan connect`
Stealth SYN Scan	1/5	Very slow	High	Root privileges	Maximum stealth	(AUTO-SET BY STEALTH OPTIONS)

Advantages & Disadvantages:

Method	Advantages	Disadvantages
SYN	Fast, stealthier, works on most targets	Needs root, can be detected by some firewalls
Connect	Works without root, reliable, works through NAT/proxies	Slower, more logs, higher detection
Stealth SYN	Extremely stealthy	Very slow, requires advanced config

UDP SCANNING

Scan	Detection	Speed	Reliability	Best For	Command
UDP (-sU)	2/5	Very slow	Variable	Discovering UDP services	`scan udp`

Common UDP Ports:

- 53 – DNS
- 67/68 – DHCP
- 69 – TFTP
- 123 – NTP
- 161 – SNMP
- 500 – IPSec

HOST DISCOVERY METHODS:

Method	Detection	Scope	Description	Command
Ping Scan (-sn)	1/5	Any network	Detects live hosts	`scan ping`
No Ping (-Pn)	−1	Any network	Skips host discovery	`no-ping`
ARP Ping (-PR)	0	Local network only	Uses ARP requests	`arp-ping`

ADVANCED SCANNING:

Method	Detection	Purpose	Command
Version Detection	2/5	Identify service versions	`scan version`
OS Detection	3/5	Identify operating systems	`scan os`
Aggressive Scan (-A)	4/5	Max info (OS + version + scripts + trace)	`scan aggressive`

7. STEALTH AND EVASION

TIMING TEMPLATES

Template	Detection Level	Speed & Delay	Use Case	Command
T0 – Paranoid	0/5	Extremely slow (5+ min between probes)	Maximum stealth, IDS evasion	`stealth timing 0`
T1 – Sneaky	1/5	Very slow (15 sec between probes)	Avoid simple detection	`stealth timing 1`
T2 – Polite	1/5	Slow but reasonable	Minimize bandwidth usage	`stealth timing 2`
T3 – Normal (Default)	2/5	Standard timing	General purpose	`stealth timing 3`
T4 – Aggressive	3/5	Fast scanning	Time-critical, fast networks	`stealth timing 4`
T5 – Insane	4/5	Very fast (may reduce accuracy)	Quick scans on fast networks	`stealth timing 5`

Characteristics per Timing Level:

- **T0:** One port at a time, huge delays, minimal impact, very hard to detect
- **T1:** Conservative, good for slow/unstable networks, avoids rate limits
- **T2:** Low bandwidth, good for large scans
- **T3:** Balanced speed/accuracy, default Nmap setting
- **T4:** Requires fast/reliable network, higher detection risk
- **T5:** Max speed, high detection risk, may miss results

PACKET MANIPULATION TECHNIQUES

Technique	Detection Level	Purpose	Command
Packet Fragmentation (-f)	-1	Evade simple packet filters	`stealth fragment`
Custom MTU (--mtu)	-1	Use specific packet sizes	`stealth mtu <size>`

Packet Fragmentation – How it Works:

- Splits TCP header across multiple packets
- Bypasses simple stateless filters
- Increases filter processing load
- Makes pattern matching harder

Custom MTU – Common Values:

- 1500: Standard Ethernet
- 1492: PPPoE
- 576: Minimum IPv4 MTU
- 1280: Minimum IPv6 MTU

IP SPOOFING & DECOYS

Technique	Detection Level	Purpose	Command
Decoy Scanning (-D)	-2	Hide real source IP among fake IPs	`stealth decoy <ip1,ip2,ME>`
Source Port Spoofing	-1	Use trusted source ports	`stealth source-port <port>`
MAC Address Spoofing	-1	Mask hardware address (local only)	`stealth spoof-mac <mac>`

Trusted Ports: 53 (DNS), 80 (HTTP), 443 (HTTPS), 25 (SMTP)

TIMING & RATE CONTROL

Technique	Detection Level	Purpose	Command
Scan Delays (--scan-delay)	-2	Add delay between probes	`stealth delay <time>`
Host Randomization	-1	Scan in random order	`stealth randomize`

Time Formats: `5s`, `500ms`, `1m`

COMBINING TECHNIQUES FOR MAXIMUM STEALTH

```
VADER> target 10.0.0.0/24
VADER> scan syn
VADER> stealth timing 1
VADER> stealth fragment
VADER> stealth delay 30s
VADER> stealth decoy 10.0.0.5,10.0.0.10,ME
VADER> stealth source-port 53
VADER> no-ping
VADER> stealth randomize
VADER> execute
```

EVASION STRATEGY BY ENVIRONMENT

Environment	Recommendations
Corporate Networks	Timing T1–T2, source port 53/80, small delays, avoid vuln scripts during business hours
Heavily Monitored	Timing T0, multiple techniques, very small ranges, long delays
Cloud Environments	Expect automation, use decoys, vary timing, watch for defensive responses

8. SERVICE ENUMERATION

WEB APPLICATION SCANNING

Service	Detection Level	Command	Purpose
HTTP/HTTPS	+2	`scripts web`	Enumerate web services & info

Common Scripts:

- `http-enum`: Directory & file enumeration
- `http-headers`: Server header analysis
- `http-methods`: Allowed HTTP methods
- `http-title`: Extract page title

SSL/TLS ASSESSMENT

- `ssl-cert` – Certificate info
- `ssl-enum-ciphers` – Cipher suite list
- `sslv2-drown` – DROWN check
- `ssl-heartbleed` – Heartbleed check

WINDOWS/SMB ENUMERATION

Service	Detection Level	Command	Purpose
SMB	+2	`scripts smb`	Enumerate Windows SMB services

Common Scripts:

- `smb-enum-shares` – List file shares
- `smb-enum-users` – Enumerate users
- `smb-os-discovery` – OS & domain info

SSH ENUMERATION

Service	Detection Level	Command	Purpose
SSH	+1	`scripts ssh`	Fingerprint SSH service

Scripts:

- `ssh-hostkey` – Host key fingerprints
- `ssh2-enum-algos` – Supported algorithms

FTP ENUMERATION

Service	Detection Level	Command	Purpose
FTP	+1	`scripts ftp`	FTP service analysis

Scripts:

- `ftp-anon` – Anonymous login check
- `ftp-banner` – Service banner info

DNS ENUMERATION

Service	Detection Level	Command	Purpose
DNS	+1	`scripts dns`	DNS server enumeration

Scripts:

- `dns-zone-transfer` – Zone transfer test
- `dns-service-discovery` – SRV records enumeration

DATABASE ENUMERATION

MySQL:

- `mysql-info` – Version & status
- `mysql-enum` – DB/user enumeration
- `mysql-brute` – Authentication test

MSSQL:

- `ms-sql-info` – Version/config info
- `ms-sql-enum` – DB enumeration
- `ms-sql-brute` – Authentication test

EMAIL SERVICE ENUMERATION

SMTP:

- `smtp-enum-users` – User enumeration
- `smtp-commands` – Command list
- `smtp-open-relay` – Relay test

POP3/IMAP:

- `pop3-capabilities`, `imap-capabilities`
- `pop3-brute`, `imap-brute`

CUSTOM SCRIPT USAGE

Specify Individual Scripts:

```
VADER> scripts http-title
VADER> scripts ssl-cert
VADER> scripts smb-enum-shares
```

Script Categories:

- `safe` – Low risk
- `intrusive` – May affect target
- `vuln` – Vulnerability checks
- `exploit` – Exploit scripts
- `auth` – Authentication
- `brute` – Brute force
- `discovery` – Info gathering

9. OUTPUT AND REPORTING

OUTPUT FORMATS

Format	Command	Best For	Characteristics
Normal Text (-oN)	`output normal <filename>`	Manual analysis, documentation	Human-readable, includes timing & summary info
XML (-oX)	`output xml <filename>`	Tool integration, automation	Machine-readable, complete scan info
Grepable (-oG)	`output grepable <filename>`	Quick filtering, scripting	One line per host, easy to grep/filter
All Formats (-oA)	`output all <basename>`	Maximum compatibility	Creates `.nmap`, `.xml`, `.gnmap` files

Example – Normal Text Output:

```
Nmap scan report for 192.168.1.1
Host is up (0.00s latency).

PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.4 (protocol 2.0)
80/tcp open  http    Apache httpd 2.4.6
```

Grepable Usage Examples:

```
grep "22/open" scan_results.gnmap
grep "80/open" scan_results.gnmap | awk '{print $2}'
```

REPORT GENERATION

Simple Report Command:

```
report <input_file> [output_file]
```

Features:

- Extracts key scan info
- Human-readable summaries
- Lists open ports/services
- Highlights important findings

Report Sections:

1. Scan summary & metadata
2. Host discovery results
3. Open ports & services
4. Service version info
5. Security observations

Advanced Reporting Tools:

- Convert XML to HTML:
  `xsltproc scan_results.xml -o report.html`
- Parse XML with Python/Ruby
- Generate PDF with LaTeX
- Integrate with ticketing systems

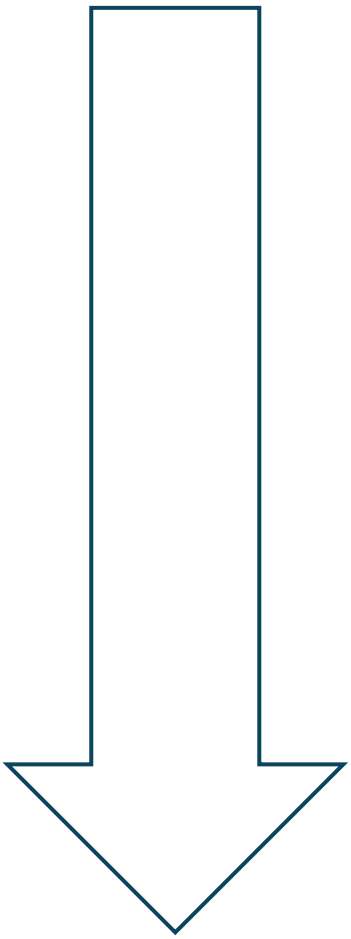
VERBOSE OUTPUT CONTROL

Level	Command	Information Included
Standard (-v)	`verbose` / `verbose 1`	Basic progress, host discovery, port scan status
Extra Verbose (-vv)	`verbose 2`	Detailed timing, individual probe results, script details
Packet Trace	`packet-trace`	All packets sent/received, full debugging

Warning: Packet tracing creates extremely large output files.

OUTPUT MANAGEMENT BEST PRACTICES

- **File Naming:**
  `company_internal_scan_2024-01-15`
  `external_web_assessment_20240115`
- **Organizing Results:**
  - By Date
  - By Target
  - By Scan Type
  - By Scope (Internal/External)
- **Retention & Security:**
  - Archive old results
  - Maintain historical data
  - Secure sensitive files
  - Regular cleanup of temp dat



10. QUICK SCAN PROFILES

AVAILABLE PROFILES

Profile	Purpose	Detection Level	Command	Configuration Summary	Typical Time
Quick	Fast initial reconnaissance	2/5	`profile quick`	TCP SYN (-sS), top 100 ports (-F), T3 timing, no scripts	1–5 min/host
Stealth	Maximum stealth/evasion	1/5	`profile stealth`	TCP SYN (-sS), T1 timing, packet fragmentation (-f), 10s delay, random hosts	30+ min/host
Comprehensive	Full system assessment	4/5	`profile full`	SYN (-sS), service/version detect (-sV), OS detect (-O), NSE default (-sC), all ports (-p-)	2+ hrs/host
Vulnerability	Security vulnerability assessment	3/5	`profile vuln`	SYN (-sS), version detect (-sV), vuln scripts, T3 timing	30–60 min/host
Discovery	Network mapping & host discovery	1/5	`profile discovery`	Ping scan (-sn), no ports, host discovery only	Seconds–minutes

PROFILE CUSTOMIZATION

Modify an Existing Profile:

```
VADER> profile stealth
VADER> ports 80,443,22
VADER> scripts web
VADER> build
```

Create a Custom Workflow:

```
VADER> profile quick
VADER> stealth timing 2
VADER> scripts safe
VADER> output xml results.xml
```

```
VADER> execute
```

PROFILE SELECTION GUIDELINES

By Environment Type:

Environment	Recommendation
Corporate Networks	Start with Quick, add stealth timing, use service-specific scripts
Cloud Environments	Use Stealth or Quick, avoid aggressive timing, watch for automation
DMZ/External	Start with Discovery, follow with Vulnerability, focus on web
Internal Networks	Use Full for asset inventory, Vuln for security, Stealth for sensitive

By Time Constraints:

- **Quick (<30 min):**
  `profile quick`, ports `80,443,22,21`, scripts `safe`
- **Comprehensive (2+ hrs):**
  `profile full`, scripts `vuln`, output `all comprehensive_scan`
- **Ongoing Monitoring:**
  Discovery daily, Vuln weekly, Full monthly

VADER – PRACTICAL EXAMPLES QUICK REFERENCE

#	Scenario	Commands	Key Points / Output

1	**Basic Network Discovery**	`target 192.168.1.0/24`
		`profile discovery`
		`output normal network_discovery.txt`
		`execute`
			Results: Responsive hosts, basic info, topology, low detection risk. **Follow-up:** Document assets, ID critical systems, plan detailed assessment.

2	**Web Application Assessment**	`target webserver.company.com`
		`scan version`
		`ports 80,443,8080,8443,8000,8888`
		`scripts web`
		`scripts safe`
		`output xml web_assessment.xml`
		`verbose 2`
		`execute`
			Info: Web server versions, dirs/files, SSL/TLS config, HTTP methods, security issues. **Analysis:** Check vulns, SSL/TLS, admin interfaces, document for further testing.

3	**Stealth Reconnaissance**	`target target-company.com`
		`scan syn`
		`stealth timing 1`
		`stealth fragment`
		`stealth delay 30s`
		`stealth decoy 1.2.3.4,5.6.7.8,ME`
		`stealth source-port 53`
		`ports 80,443,22,25,53,110,143,993,995`
		`no-ping`
		`stealth randomize`
		`output all stealth_recon`
		`execute`
			Stealth: T1 timing, fragmentation, long delays, decoys, DNS port, no ping, randomized order. **Duration:** Hours. **Risk:** Minimal.

4	**Comprehensive Security Assessment**	`target 10.0.0.100-110`
		`profile full`
		`scripts vuln`
		`scripts safe`
		`add-reason`
		`verbose 2`
		`output all security_assessment_$(date +%Y%m%d)`
		`execute`
			Coverage: All ports, version detection, OS ID, vuln detection, safe scripts, reasoning for ports. **Results:** Service inventory, vulns, config issues, recommendations.

5	**Database Service Enumeration**	`target database-subnet.company.com`
		`scan version`
		`ports 1433,3306,5432,1521,27017,6379`
		`scripts safe`
		`stealth timing 2`
		`output normal database_enum.txt`
		`execute`
			Services: MSSQL, MySQL, PostgreSQL, Oracle, MongoDB, Redis. **Security:** Test default creds, info leaks, unencrypted

			conns, access controls.

6	**Windows Environment Assessment**	`target 192.168.10.0/24`
		`scan version`
		`ports 135,139,445,389,636,3268,3269,88,53`
		`scripts smb`
		`scripts safe`
		`stealth timing 2`
		`output xml windows_assessment.xml`
		`execute`
			Targets: RPC, SMB, LDAP, GC, Kerberos, DNS. **Info:** Domain controllers, shares, users, trust rels, OS versions.

7	**IoT Device Discovery**	`target 192.168.100.0/24`
		`scan syn`
		`ports 80,443,8080,23,22,21,161,502,20000,30000`
		`scripts safe`
		`scripts web`
		`stealth timing 2`
		`output normal iot_discovery.txt`
		`execute`
			Ports: Web (80/443/8080), Telnet, SSH, FTP, SNMP, Modbus, custom high ports. **Security:** Default creds, unencrypted protocols, outdated firmware, misconfigs.

8	**Cloud Infrastructure Assessment**	`target cloud-app.company.com`
		`scan version`
		`ports 80,443,22,3389,1433,3306,5432,6379,27017`
		`scripts web`
		`scripts safe`
		`stealth timing 3`
		`stealth randomize`
		`output xml cloud_assessment.xml`
		`verbose on`
		`execute`
			Cloud Notes: Auto defenses, rate limits, Geo-IP blocks, DDoS protection. **Strategy:** Moderate timing, watch defenses, focus public services, log provider hints.

9	**Vulnerability Management Workflow**	`target file production_servers.txt`
		`profile vuln`
		`scripts safe`
		`stealth timing 2`
		`output all vuln_scan_$(date +%Y%m%d_%H%M)`
		`verbose on`
		`execute`
			Workflow: Pre-scan targets → run vuln scan → generate report → import to tracker → compare trends.

10	**Red Team Reconnaissance**	`target target-organization.com`
		`profile stealth`
		`stealth decoy 10.0.0.5,10.0.0.10,ME,10.0.0.15`
		`stealth source-port 53`
		`ports 80,443,22,25,53,110,143`
		`scripts safe`
		`output normal redteam_recon_$(date +%Y%m%d).txt`
		`execute`
			Red Team: Minimal footprint, realistic sim, persistence planning, intel focus. **OpSec:** Use proxies, vary timing, watch defenses, keep logs.

12. BEST PRACTICES

PRE-SCAN PREPARATION

LEGAL AND AUTHORIZATION

1. **Written Authorization:** Always obtain explicit written permission before scanning.
2. **Scope Definition:** Clearly define target networks and systems.
3. **Time Windows:** Agree on acceptable scanning windows.
4. **Emergency Contacts:** Maintain contact information for target administrators.
5. **Incident Response:** Have procedures for unexpected issues.

TECHNICAL PREPARATION

1. **Network Reconnaissance:** Understand target network topology.
2. **Baseline Establishment:** Document normal network behavior.
3. **Tool Verification:** Test VADER functionality in a safe environment.
4. **Resource Planning:** Ensure adequate time and computational resources.
5. **Backup Plans:** Prepare alternative scanning approaches.

DOCUMENTATION REQUIREMENTS

1. **Scan Planning:** Document objectives and methodology.
2. **Target Lists:** Maintain accurate asset inventories.
3. **Configuration Records:** Save VADER command configurations.
4. **Timeline Documentation:** Record scan schedules and durations.

SCAN EXECUTION GUIDELINES

TIMING AND SCHEDULING

1. **Business Hours:** Avoid peak hours unless authorized.
2. **Maintenance Windows:** Schedule intensive scans during maintenance periods.
3. **Gradual Escalation:** Start with minimal scans, increase intensity gradually.
4. **Break Intervals:** Include breaks in long scanning sessions.

DETECTION MANAGEMENT

1. **Start Stealthily:** Begin with low-detection-level techniques.
2. **Monitor Responses:** Watch for defensive reactions.
3. **Adjust Accordingly:** Modify approach based on observed responses.
4. **Document Reactions:** Record any defensive measures encountered.

NETWORK CONSIDERATIONS

1. **Bandwidth Impact:** Monitor and limit network usage.
2. **Target Stability:** Avoid overwhelming unstable systems.
3. **Service Disruption:** Minimize risk of service interruption.
4. **Error Handling:** Implement proper error handling and recovery.

RESULTS ANALYSIS

DATA VALIDATION

1. **Result Verification:** Confirm scan results through multiple methods.
2. **False Positive Identification:** Distinguish genuine findings from false positives.
3. **Context Analysis:** Consider results within business and technical context.
4. **Trend Analysis:** Compare results with historical data.

SECURITY ASSESSMENT

1. **Vulnerability Prioritization:** Rank findings by risk level.
2. **Impact Assessment:** Evaluate potential business impact.
3. **Exploitability Analysis:** Assess ease of exploitation.
4. **Mitigation Planning:** Develop remediation recommendations.

DOCUMENTATION STANDARDS

1. **Comprehensive Records:** Document all findings thoroughly.
2. **Evidence Collection:** Maintain proof of discovered vulnerabilities.
3. **Reproducible Results:** Provide sufficient detail for verification.
4. **Executive Summaries:** Create management-appropriate summaries.

REPORTING AND COMMUNICATION

REPORT STRUCTURE

1. **Executive Summary:** High-level overview for management.
2. **Technical Details:** Comprehensive technical findings.
3. **Risk Assessment:** Business impact analysis.
4. **Recommendations:** Specific remediation steps.
5. **Appendices:** Supporting data and evidence.

COMMUNICATION PROTOCOLS

1. **Immediate Notifications:** Alert for critical vulnerabilities.
2. **Regular Updates:** Provide progress reports during assessments.
3. **Final Reporting:** Deliver comprehensive final reports.
4. **Follow-Up Reviews:** Schedule remediation verification.

STAKEHOLDER MANAGEMENT

1. **Technical Teams:** Detailed technical communications.
2. **Management:** Risk-focused business communications.
3. **Compliance:** Regulatory requirement reporting.
4. **Vendors:** Coordinated disclosure when appropriate.

TOOL MANAGEMENT AND MAINTENANCE

CONFIGURATION MANAGEMENT

1. **Version Control:** Track VADER script versions.
2. **Configuration Backups:** Save and version command configurations.
3. **Environment Consistency:** Maintain consistent scanning environments.
4. **Update Procedures:** Establish regular update processes.

SECURITY CONSIDERATIONS

1. **Access Control:** Restrict VADER access to authorized personnel.
2. **Audit Trails:** Maintain logs of tool usage.
3. **Data Protection:** Secure scan results and configurations.
4. **Incident Response:** Prepare for potential tool compromise.

PERFORMANCE OPTIMIZATION

1. **Resource Monitoring:** Track system resource usage during scans.
2. **Network Optimization:** Optimize network configuration for scanning.
3. **Parallel Processing:** Implement efficient parallel scanning.
4. **Result Storage:** Implement efficient data storage and retrieval.

13. TROUBLESHOOTING

COMMON ISSUES AND SOLUTIONS

INSTALLATION PROBLEMS

Issue	Error Message	Solutions
nmap not found in PATH	`nmap is not installed or not in PATH`	1. Install nmap using your package manager. 2. Verify installation: `which nmap`. 3. Add nmap directory to PATH. 4. Use full path to the nmap binary.
Permission denied errors	`Permission denied when executing VADER`	1. Check file permissions: `ls -la vader.sh`. 2. Make it executable: `chmod +x vader.sh`. 3. Run with appropriate privileges. 4. Check directory permissions.
Root privileges required	NOT RUNNING AS ROOT. SOME SCAN TYPES MAY NOT WORK.	1. Run with `sudo ./vader.sh`. 2. Use connect scans instead of SYN scans. 3. Configure capabilities for the nmap binary. 4. Use unprivileged scanning techniques.

SCANNING ISSUES

No response from targets (HOST APPEARS TO BE DOWN)

- Verify target reachability: `ping <target>`
- Check firewall rules.
- Use `-Pn` to skip host discovery.
- Try different timing templates.
- Verify network connectivity.

Scan taking too long

- Use faster timing (`-T4` or `-T5`).
- Reduce port range.
- Skip host discovery.
- Increase parallelism.

- Focus on specific targets.

High number of filtered ports

- Firewall may be blocking probes.
- Try different scan types (SYN vs Connect).
- Use source port spoofing.
- Apply packet fragmentation.
- Adjust timing to avoid rate limiting.

NETWORK AND CONNECTIVITY ISSUES

Problem	Diagnostic Steps / Solutions
Network timeouts and errors	1. `traceroute <target>` 2. Check DNS: `nslookup <hostname>` 3. `ping <target>` 4. `route -n` 5. Check `ifconfig`
Rate limiting encountered	1. Slow down timing. 2. Add delays. 3. Rotate source IPs. 4. Smaller batch sizes. 5. Spread scans over time.

RESULT INTERPRETATION ISSUES

Unexpected port states

- Manually test connection.
- Use `--reason` to understand state.
- Compare different scan types.
- Check for load balancers/proxies.
- Consider NAT effects.

Inconsistent results between scans

- Services may be dynamic.
- Load balancing can affect results.
- IDS/IPS may interfere.
- Network conditions may change.
- Firewall rules could be updated.

PERFORMANCE OPTIMIZATION

Scan Speed Optimization

- Use appropriate timing templates.
- Limit port selection.
- Skip unnecessary host discovery.
- Run scans in parallel.
- Group targets efficiently.

Resource Usage

- Monitor CPU & RAM usage.
- Optimize NIC configuration.
- Use efficient data handling.
- Balance accuracy vs. speed.

Network Optimization

- Manage bandwidth.
- Apply rate limiting.
- Schedule scans off-hours.
- Use QoS if possible.

ERROR MESSAGES AND SOLUTIONS

Error Message	Resolution
`Target validation failed`	Check IP/hostname format, DNS resolution, CIDR notation, and typos.
`Command execution failed`	Verify nmap installation, syntax, system resources, and logs.
`Permission denied for port scanning`	Run as root, use connect scans, configure capabilities, check firewall rules.
`Network unreachable`	Verify connectivity, routing, and target existence. Test with ping.

DEBUG MODE

Commands for troubleshooting:

```
packet-trace    # Enable packet-level debugging
verbose 2       # Maximum verbosity
add-reason      # Show reasons for port states
```

LOG ANALYSIS

- **Location:** `vader_errors.log`
- **Format:** Timestamped error messages
- **Purpose:** Identify patterns and recurring issues

GETTING HELP AND SUPPORT

Built-in Help System

- General Help: `help`
- Specific Topics: `explain <flag>`
- Examples: `examples <type>`
- Detection Info: `detection`

Community Resources

- Nmap Documentation
- Security Forums
- Ethical Hacking Courses
- InfoSec Professional Groups

Professional Support

- Penetration Testing Services
- Security Consultants
- Hands-on Training Providers
- Certification Bodies

14. LEGAL AND ETHICAL CONSIDERATIONS

1. LEGAL FRAMEWORK

Category	Details
Authorization Requirements	- **Written Permission:** Always obtain explicit, written authorization before conducting any network scanning or security assessment activities. - **Possible Violations:** • Computer Fraud and Abuse Act (USA) • Computer Misuse Act (UK) • Criminal Code (various countries) • Corporate policies & terms of service • Industry-specific regulations
Scope Documentation	Clearly define: • Authorized IP ranges / hostnames • Allowed scanning techniques & intensity • Approved time windows • Emergency contact procedures • Incident response protocols

2. COMPLIANCE CONSIDERATIONS

Standard / Regulation	Area of Application
PCI DSS	Payment card industry security
HIPAA	Healthcare data protection
SOX	Financial reporting compliance
ISO 27001	Information security management
NIST CSF	Cybersecurity risk management
GDPR	EU data protection law
CCPA	California data privacy law
Local Laws	Country/state-specific cybersecurity rules
International Restrictions	Cross-border scanning limitations

3. LEGAL RISK MANAGEMENT

Documentation Requirements

1. Keep detailed logs of scanning activities
2. Record authorization & approvals
3. Save scan configurations and methods
4. Preserve compliance-related evidence
5. Securely store sensitive data

Liability Protection

1. Obtain professional liability insurance
2. Include protective contract clauses
3. Follow industry-standard methodologies
4. Maintain professional certifications
5. Consult legal counsel when needed

4. ETHICAL GUIDELINES

A. RESPONSIBLE DISCLOSURE

Step	Action
1	Notify system owners of critical vulnerabilities immediately
2	Agree on reasonable remediation timelines
3	Share details only with authorized parties
4	Keep documentation of all communications
5	Verify remediation completion

B. CONFIDENTIALITY

1. Use secure data handling procedures
2. Restrict access to reports and findings
3. Define retention periods
4. Securely dispose of sensitive data
5. Adhere to NDAs and confidentiality clauses

5. MINIMIZING IMPACT

System Stability	Business Operations
Start with least intrusive scans	Schedule during maintenance windows

Monitor for instability	Limit bandwidth/resource usage
Have backup/recovery plans	Avoid service-disrupting methods
Keep communication open	Coordinate with IT teams
Stop scans if issues occur	Prepare recovery procedures

6. BEST PRACTICES FOR ETHICAL SCANNING

PRE-ENGAGEMENT

- **Stakeholder Engagement:** Coordinate with sponsors, IT/security teams, and legal departments; verify insurance; set emergency protocols
- **Risk Assessment:** Evaluate technical, business, legal, and reputational risks; define mitigation strategies

DURING ENGAGEMENT

- **Monitoring & Communication:** Progress updates, escalation procedures, emergency contacts, system monitoring
- **Quality Assurance:** Follow methodologies, verify tools, cross-check findings, peer reviews, maintain documentation

POST-ENGAGEMENT

- **Reporting:** Timely delivery, accurate findings, risk prioritization, remediation advice, executive briefings
- **Data Management:** Secure storage, strict access control, compliance retention, secure disposal, maintain audit trails

7. INDUSTRY STANDARDS AND FRAMEWORKS

PROFESSIONAL CERTIFICATIONS

Certification	Focus Area
CEH	Ethical hacking
OSCP	Advanced penetration testing
CISSP	Information security governance

CISA	Information systems auditing
GPEN	GIAC penetration testing

Continuing Education

1. Stay updated on tools and techniques
2. Attend training and conferences
3. Engage with security communities
4. Follow legal/regulatory changes

TESTING METHODOLOGIES

Standard	Description
OWASP	Web application security testing guidelines
NIST SP 800-115	Technical guide for security testing
PTES	Penetration Testing Execution Standard
OSSTMM	Open Source Security Testing Methodology Manual
ISSAF	Information Systems Security Assessment Framework

Organizational Standards

- Develop internal procedures
- Implement quality control
- Peer review processes
- Standardized reporting formats
- Maintain methodology documentation

THIS PROJECT REPRESENTS A SIGNIFICANT CONTRIBUTION TO THE CYBERSECURITY COMMUNITY, COMBINING TECHNICAL EXCELLENCE WITH EDUCATIONAL VALUE TO CREATE A TOOL THAT MAKES ADVANCED NETWORK SECURITY ASSESSMENT ACCESSIBLE TO PROFESSIONALS AT ALL LEVELS.