



# Tracing Attacks through Non-Cooperating Networks

## Designing and Implementing Traceback Mechanisms That Can Bridge “Uncooperative Gaps”

### Problem

As the Intelligence Community (IC) shares its information electronically across agencies and with ad hoc coalition partners, both the risk of compromise and the need for effective attribution of cyber attacks increase dramatically. The current state of the art in attack attribution relies heavily on the cooperation of neighboring network nodes along potential attack paths to monitor attack traffic and trace it through relay and branch points. In practice, however, attacks often wind their way through networks that are outside the defenders' control, or through nodes within the defenders' networks that may have been rendered uncooperative through failure, prior compromise, or insider action. In these important and common cases, new attack traceback techniques are required to enable definitive attack attribution.

### Approach

SPARTA's Security Research Division, and Boeing Phantom Works have designed, and evaluated novel techniques for tracing attacks back through uncooperative networks to their origins. In addition, we have combined these techniques synergistically with previously developed advanced cooperative traceback capabilities, resulting in a unified, coordinated traceback architecture. We have explored a range of novel traceback techniques, and selected the most promising for further development.

The broad vision of this work is to develop a library of advanced traceback techniques to be applied where appropriate, depending on network topology, availability of monitoring components, ambient traffic characteristics, attack class, and traceback timeliness, stealthiness, and certainty requirements. This vision includes leveraging our existing traceback architecture that can integrate these new

techniques with conventional cooperative traceback techniques to provide end-to-end traceback and attribution as accurately as possible given the constraints of the network environment. In this architecture, intrusion detection components can automatically trigger traceback initiation as well as automated deployment of defensive network traffic filters along the attack path. We are producing prototype implementations of these new traceback techniques and evaluating them through analysis, simulation, and experimentation individually and as part of an integrated, larger-scale experimental deployment.

### Solution

A comprehensive traceback capability must address the problem of transforming relays in the attack path, which modify and redirect attack traffic and can thwart traceback attempts. These relay nodes may include anonymizing proxies, firewalls, network address translation devices, “reflectors” exploited in distributed denial-of-service attacks, and compromised hosts used as stepping-stones. (See figure.) In a broad range of circumstances, we can provide the equivalent of cooperative traceback through uncooperative nodes, including transforming relays, by leveraging the capabilities of cooperating neighbor nodes. These cooperating neighbor nodes (1) passively monitor traffic, (2) aggregate traffic into likely application-layer groups (e.g. by TCP/IP five-tuple), (3) summarize these traffic aggregates by various statistical means, and (4) compare

We have developed and evaluated prototypes of three protocol-independent traffic analysis methods: idle-active transition profiling, packet-size distribution profiling, and substring content fingerprinting. These three methods extract roughly orthogonal characteristics (temporal, message size, message content) from traffic



## Tracing Attacks through Non-Cooperating Networks

Designing and Implementing Traceback Mechanisms That Can Bridge "Uncooperative Gaps"

flows. Depending on the stepping-stones employed, and the attack scenario, one or more of these methods may be effective for tracing the attack flow toward its origin.

We have also investigated two additional protocol-specific traffic analysis methods that address specific attack scenarios not well covered by our three protocol-independent methods. These methods are: TLS frame size profiling, and HTTP proxy traceback. TLS frame size profiling method addresses scenarios

involving encrypted TLS streams where the protocol-independent content fingerprinting method is ineffective. The HTTP proxy traceback method addresses scenarios where an attacker using highly loaded HTTP proxies may confound the other methods.

The table below summarizes the various techniques, identifying the data requirements of each, and the scenarios where the techniques are most effective.

Technique	Data Required		Signatures require training data?	Scenarios where effective
	Headers	Payloads		
Content fingerprinting	Yes	Yes	Yes	Plaintext application, e.g. HTTP
Packet size profiling	Yes	No	Yes	DDoS, worm propagation, port scans
Idle-to-active transition profiling	Yes	No	No	Interactive sessions, e.g. remote shell, RDP
TLS frame size profiling	Yes	Yes	No	End-to-end TLS, despite Socks, NAT, HTTP Proxy
HTTP proxy traceback	Yes	Optional	No	HTTP via proxies

## Statistical Flow Identification Techniques