Secure Collective Defense (SCOLD) Network[1]

C. Edward Chow
Yu Cai
David Wilkinson

Department of Computer Science
University of Colorado at Colorado Springs
1420 Austin Bluffs Parkway
Colorado Springs, CO 80933-7150
USA

TEL: (719)262-3110
Email: {chow, ycai, dwilkin}@cs.uccs.edu

Abstract: The increasing network attacks reveal one of the fundamental security problems of today's Internet. Many Internet services, such as DNS and routing protocols, were not originally designed with security as one of the basic requirements. It is difficult to modify the existing protocols or network architecture without significant work. At the same time, it offers an opportunity to create new, secure and reliable network protocols, and packet delivery systems. We present, in this paper, a prototype of the Secure Collective Defense (SCOLD) system that utilizes collective resources from participation organizations, tighten coordination and new cyber security defense techniques against Distributed Denial of Services (DDoS) attack. SCOLD tolerates DDoS attacks with alternate routes via a set of proxy servers with intrusion detection, and secure Domain Name System (DNS) updates. The research results and insights obtained from this project can improve the security of the networks and have a broader impact on the network architecture and the client side network software interface.

Index Terms: Intrusion Tolerance, DDoS, Secure Collective Defense, Alternate Route, Secure DNS update.

1. Introduction

The brief service disruption on the nine of the thirteen DNS root servers caused by DDoS bandwidth attacks on October 2002 [1] is one of the most prominent attacks on DNS recently. The increasing frequency and severity of network attacks reveal one of the fundamental security problems of today's Internet. Many Internet services, such as DNS and routing protocols, were not originally designed with security as one of the basic requirements. Therefore, the existing network architecture

[1] This work is based on research sponsored by the Air Force Research Laboratory, under agreement number F49620-03-1-0207. The view and conclusions contained herein are those of the authors and should not be interpreted as necessarily represented the official policies or endorsements, either expressed or implied, of the Air Force Research Laboratory or the U. S. Government."

needs to be strengthened and protocols need to be enhanced or re-designed with security as the basic consideration.

A study conducted by the University of California, San Diego, detected approximately 12,805 Denial of Service attacks against more than 5000 targets during a three-week period in mid-2001 [2]. Even CERT, the authority that warns Internet users on security threats, fell victim to DDoS in May 2001 [2]. In a recent survey conducted by the SANS Institute on ``How to Eliminate the Ten Most Critical Internet Security Threats", the number-one Internet vulnerability reported by survey participants was BIND weaknesses [3]. BIND is the open-source software package that powers the majority of Internet DNS servers [18].

The general objective of the Secure Collective Defense (SCOLD) project is to create a secure collective Internet defense system that utilizes collective resources from participation organizations, tighten coordination and new cyber security defense techniques. SCOLD will tolerate Distributed Denial of Services (DDoS) attacks with alternate routes via a set of proxy servers with intrusion detection, and secure Domain Name System (DNS) updates.

Most of the organizations have multiple gateways and can deploy multi-homing schemes using alternate gateways when the main gateway is attacked. But once the alternate gateway's IP address is revealed, it is also subject to DDoS attacks. We propose to explore the use of protected indirect routes over a collection of geographical separated proxy servers to those alternate gateways and hide the IP address of those alternate gateways from the clients. The clients or client site DNS servers will be informed of the indirect routes through secure DNS updates. The new DNS entries will contain records like (Domain name, IP address, a set of IP addresses of designated proxy servers). Client requests will be sent to one of the selected proxy servers or spread over the set of designated proxy servers for better performance with aggregated bandwidth. The proxy server will be enhanced with integrated Intrusion Detection System (IDS) and a firewall filter to block intrusion traffic that may try to come in through the indirect route. The detection of intrusion on those proxy servers provides additional identification and isolation of the source of the attacks. The collection of proxy servers can be provided by participating organizations of a consortium, or branches of an organization. The proxy servers know the IP addresses of an alternate gateway of the target site and relay the packets from the clients over an IP tunnel to the target site. An indirect route can then be set up from the clients to the attacked target through designated proxy servers and alternate gateway.

This new set of network protocols and infrastructure can be used to defend against generic DDoS attacks as well as protect the root DNS servers from DDoS attacks.

1.2 Related Work

Recent work by Angela Cearns from University of Colorado, Colorado Springs, implemented an Autonomous Anti-DDoS network (A2D2) with enhanced SNORT IDS for detecting subnet spoof attacks and with adaptive rate limiting and Class Based Queuing (CBQ) firewall rules for effective intrusion handling [4]. Steven Cheung in University of California at Davis, introduced a wrapper-based solution to protect DNS, and a detection-based message authentication scheme to protect routers [5].

Network Associates Labs and Boeing developed the Intrusion Detection and Isolation Protocol (IDIP) to support real-time tracking and containment of attacks that cross network boundaries [7]. Service Location Protocol (SLP) is an IETF protocol that provides automatic client configuration for applications and advertisement for network services, i.e. locating IDIP nodes [8]. We are going to incorporate IDIP and SLP in future SCOLD system.

J. Mirkovic, J. Martin and P. Reiher from University of California at Los Angeles provided a compressive taxonomy of DDoS attacks and DDoS Defense Mechanisms [6]. According to the classification of DDoS defense mechanisms in the taxonomy, the SCOLD falls into the category of reactive, reconfiguration and cooperative. Reactive mechanism strives to alleviate the impact of the DDoS attack on the victim instead of eliminating the attacks. Reconfiguration mechanism changes the topology of the victim or the intermediate network. Cooperative mechanism is achieved through cooperation with other entities. Related works in Reconfiguration mechanism include reconfigurable overlay networks ([21], [22]), resource replication services [23], attack isolation strategies ([24], [25]), etc. These works focused on either adding more resources to the victim or isolating the attack machines. But the SCOLD system manages to redirect client traffic through a set of proxy servers by indirect route under DDoS attacks. Cooperative mechanism is limited by the highly independent nature of Internet. The SCOLD system tries to utilize collective resources from participation organizations with tighten coordination and cooperation.

The balance of this paper is organized as follows. In Section 2, we introduce the design of SCOLD system. In Section 3, we present the design of secure DNS update. In Section 4, we present the design of indirect route through IP tunneling. Test and performance results are presented in Section 5. Our conclusions and future works are in Section 6.

2. SCOLD Design

2.1 SCOLD Overview

There are two basic approaches to defend against DDoS attacks: one is called *intrusion blocking/tracking* that actively tracks down and blocks the traffic generated from those infected machines, and identifies the mastermind intruder; the other is called *intrusion tolerance* that studies how to tolerate intrusion and provides alternate routes for legitimate clients to access the target site.

The SCOLD system deals with mainly the latter approach by designing new software modules and protocols for providing such alternate routes. But by dividing the clients to come in through different proxy servers with intrusion detection devices, the SCOLD system can also provide useful information for tracking down the intruder.

One of the critical components in the SCOLD system is the shared usage of a collection of geographically distributed proxy servers, either provided by a service provider or contributed by each participating organization of a consortium.

When a site is attacked, its intrusion detection system will generate the security alarms and send a request to the SCOLD coordinator for setting up indirect route. The coordinator then updates the DNS servers of the client sites through secure DNS updates. The clients get the IP addresses of the proxy servers, and send packets through the designated proxy servers. The designated proxy server knows the IP addresses of an alternate gateway of the target site and relays the packets from the clients over an IP tunnel to the target site. The designated proxy server is integrated with the intrusion detection system to detect and block potential DDoS attacks on these alternate gateways.

The secure DNS update utilizes the Secure Socket Layer protocol for authentication and encryption. The existing DNS servers need to be modified to save the new cache entries with the domain name and IP address of the target machine, and the IP address of the designate proxy servers. The hostname resolving library on a client machine needs to be modified to accept the new type of DNS query results. For the exploratory prototype, we propose to modify the popular BIND 9 DNS software package from Internet Software Consortium [9, 10].

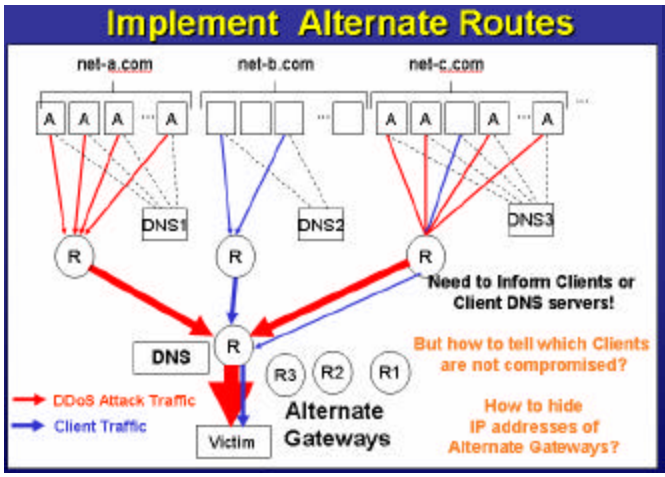

Figure 1: DDoS attack without alternate routes [11]

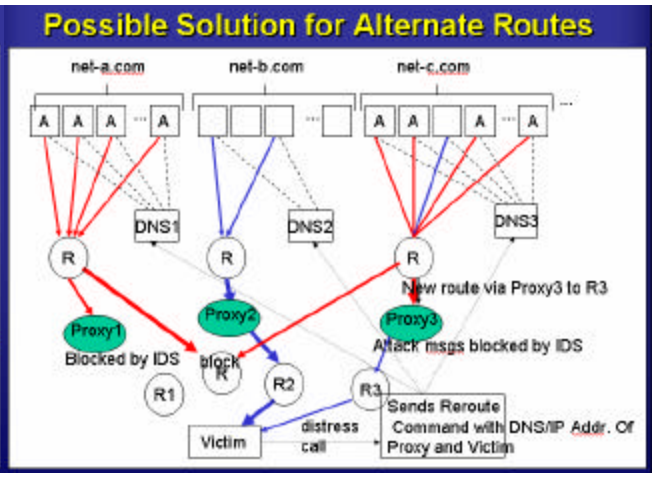

Figure 2: DDoS attack with alternate routes [11]

Figure 1 shows the target under DDoS attack, without the implementation of alternate routes. As a consequence, the bandwidth of legitimate clients is greatly reduced. Figure 2 shows the target under DDoS attack, with the implementation of alternate routes. All users will get updated alternate DNS entry information. But the attack network will be blocked at proxy servers, and the legitimate users will be re-directed to alternate gateway, then to the final destination.

2.2 Design Considerations

A. The need to hide alternate gateway

Organizations that have multiple gateways can deploy alternate gateways when the main gateway is attacked. But once the alternate gateway's IP address is revealed on internet, it is also subject to DDoS attacks. We can use protected indirect routes over a collection of proxy servers to those alternate gateways. Therefore, the proxy servers will act as the frontline against DDoS attacks, and the IP address of the alternate gateways were hide.

B. The need for indirect routing

The normal route from the client to the intended target will be severely impacted by DDoS attack. Therefore, we need to use indirect routing to route the client traffic pass through the proxy server, then the alternate gateway, and finally the target. One way for indirect routing is using IP tunneling.

C. The need for secure DNS update and secure connection

The dynamic update of DNS record needs to be secure and authenticated. Otherwise the malicious clients can send fake DNS record update and redirect the traffic of legitimate clients.

For the similar reason, certain connections between proxy servers, alternate gateways, target and coordinators need to be secure and mutually authenticated. Because of the overhead of secure connection, we need to keep the minimum usage of secure connection.

D. The advantage of using a set of proxy servers.

Using a set of proxy servers can avoid vulnerable bottleneck points in the intermediate network. If a selected proxy server gets severe DDoS attacks, the SCOLD can inform the clients to use other available proxy servers, and reset the indirect route.

A collection of geographical separated proxy servers also provide the possibility to spread the load and client request, and overcome the overhead associated with indirect routing and secure DNS update.

3. Secure DNS update

We have extend Bind9 v.9.2.2 DNS server software package, which is written and maintained by the Internet Software Consortium [18], to enable secure DNS update and alternate proxy IP address, by modifying the nsreroute command, and an add-on to the BIND9 DNS software.

In the rest of the paper, we use "Client" specifically refer to a client machine in the SCOLD system in Figure 3. Same for "Proxy", "Gateway", "Target", "Coordinator", "ClientDNS" and "TargetDNS".

The Steps for secure DNS update are as follows (Figure 3):

1. The Target gets DDoS attack, it will raise an alarm and notify the Coordinator to issue a command for secure DNS update.
2. The Coordinator has a list of available proxy servers IP addresses for the Target. The Coordinator sends message to the TargetDNS, updates its DNS records with Proxy IP addresses as ALT data type.
3. The Coordinator sends message to the Proxy.
4. The Proxy sends message to the ClientDNS to notify that an indirect route is ready to be set up.
5. The Client queries the ClientDNS, and the ClientDNS queries the TargetDNS to get DNS record. (The DNS record alive time is set to be very short to enable dynamic DNS update).
6. If the route from the ClientDNS to the TargetDNS is not severely affected by DDoS attack, the Client DNS will talk to the TargetDNS through normal Internet route and fetch the updated DNS records. Otherwise the ClientDNS needs to setup indirect route to the TargetDNS through the Proxy and the Gateway as described in Section 4.
7. Restore normal DNS record. Once the DDoS attacks stop, the Target will notify the Coordinator and ask it to issue commands to restore normal DNS record with similar steps in steps (1-6).

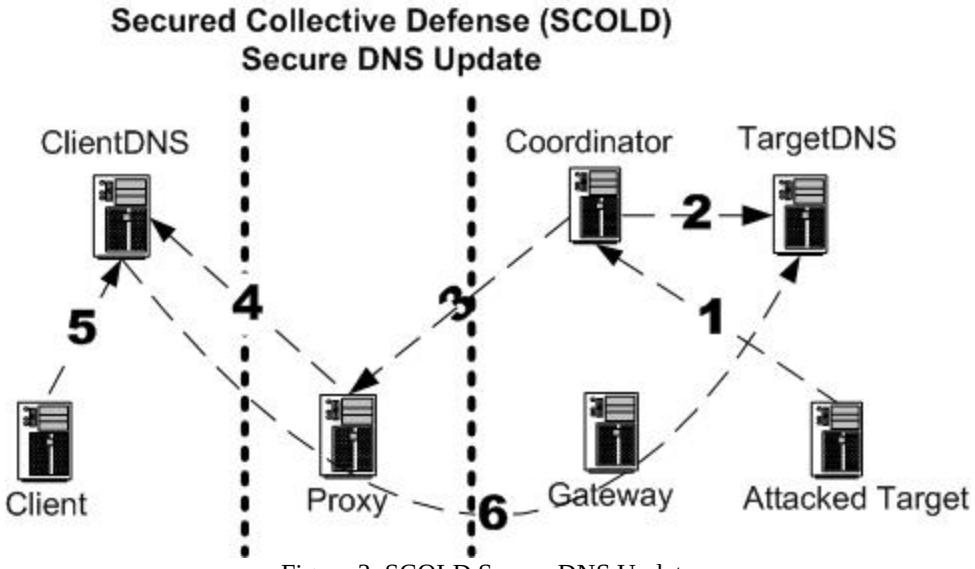

Figure 3: SCOLD Secure DNS Update

This updated zone file with alternate proxy IP addresses will look like the following:

```
target.targetnet.com. 10 IN A 133.41.96.71
target.targetnet.com. 10 IN ALT 203.55.57.102
                      10 IN ALT 203.55.57.103
                      10 IN ALT 185.11.16.49
                      10 IN ALT 221.46.56.38
```

After the secure DNS update, the client will get a DNS record with multiple proxy server IP addresses as ALT type in addition to normal domain name and IP address. An indirect route can then be setup from client machine through the selected proxy server.

We use OpenSSL to authenticate and encrypt the control message communicated between DNS server and the SCOLD coordinator. [12]

4. Indirect Route

We have investigated several alternatives for implementing indirect routing. The alternatives include Zebedee[15], SOCKS proxy server[16], and IP tunnel[17].

IP tunnel (also called IP encapsulation) is a technique to encapsulate IP datagram within IP datagrams. This allows datagrams destined for one IP address to be wrapped and redirected to another IP address. The IP tunnel can be set up from Linux to Linux, windows to windows, or between Linux and windows (windows must be Windows 2000 server and above).

The advantage of using IP tunnel is that it is a layer three protocol; therefore all the upper layer protocols and applications can use it transparently. Second, IP tunnel is installed with Redhat Linux 8 and 9 by default, and setting up IP tunnel doesn't require additional software installation or kernel compilation. The other advantage of IP Tunnel is that IP tunnel is essentially a device descriptor, it consumes limited system resources on the server, therefore the number of IP tunnels available is primitively limited by server resources.

Even through IP Tunnel supports any upper layer transport protocols, it increases overhead by an extra set of IP headers. Typically this is 20 bytes per packet, so if the normal packet size (MTU) on a network is 1500 bytes, a packet that is sent through a tunnel can only be 1480 bytes big, therefore the payload size is reduced. This also causes fragmentation and reassembly overhead. But these overheads can be reduced or avoided by setting smaller MTU at the client side.

4.1 Indirect Route by Modifying Client's Resolve Library

The resolve library is a shared library located usually in /usr/lib or /lib directory in Redhat Linux, named as libresolv-nnn.so (nnn is the version). Also there are a couple of soft link to libresolv-nnn.so which named as libresolv-nnn.so.1, libresolv-nnn.so.2 in the same directory. The resolve library is a dynamically loadable library which is usually called when client queries the target by domain name. The resolve library will then query the DNS and return the corresponding IP address of target.

The resolve library is usually distributed as part of the glibc package [13]. Since resolve library is part of the system library, be extremely careful when modify, compile and install the library. A small error can easily make the system unstable or even crash it. We modified one of the resolve library routines named res_query() to enable indirect route. The resolve library we modified is version 2.3.2, and is tested on Linux Redhat 8 and 9.

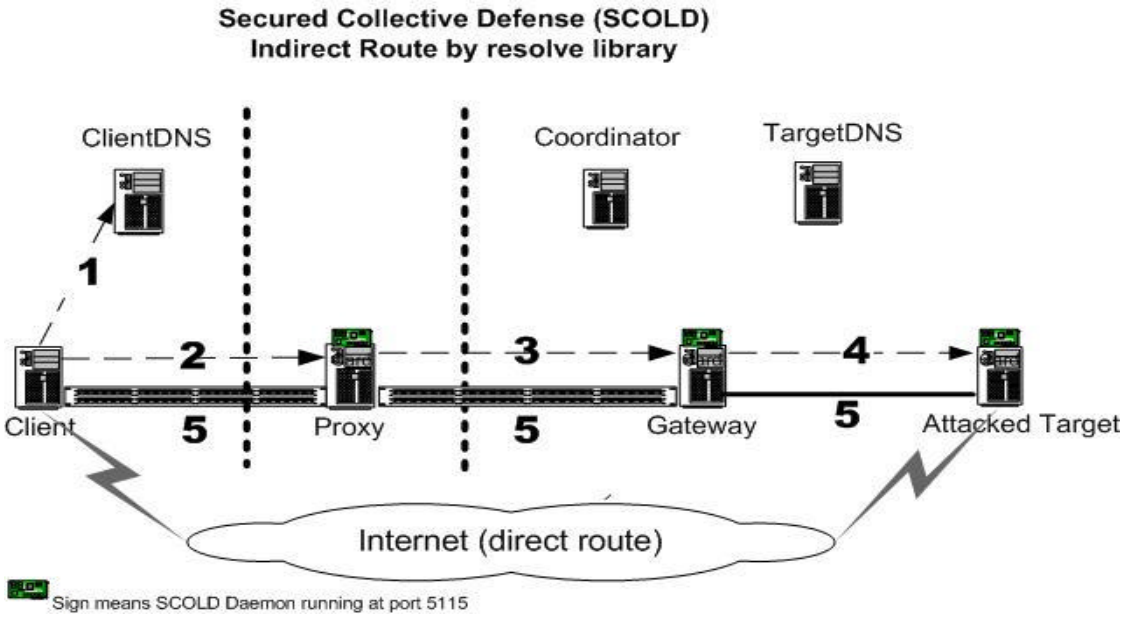

Figure 4: SCOLD Indirect Route by Modifying Resolve Library

The steps of setting up indirect route using resolve library are as follows (Figure 4):

1. The Client queries the Target by hostname, and the resolve library on the Client will query the ClientDNS for name resolution. The Client will get updated DNS information with proxy IP addresses beside normal Target IP address.
2. In resolve library, the Client will choose one of the Proxy IP addresses, send a message to that Proxy, and setup IP tunnel from the Client to the Proxy. If the Proxy fails to response, the Client will choose the next available Proxy IP.
3. The Proxy will choose one of the alternate gateways, send a message to that gateway; also, the Proxy will re-set its routing table and firewall rules, setup the IP tunnels to Client and to Gateway.
4. The Gateway will send a message to the Target, re-set the routing table and firewall rules on the Gateway, and setup the IP tunnel to Proxy. After getting the message from the Gateway, the Target will re-set its routing table and firewall rules to enable indirect route.
5. The indirect route is set up. The Client can access the Target through indirect route.
6. Clean up the indirect route: When the DDoS attacks stop, through the secure DNS update, the clientDNS will get normal IP/hostname without proxy IP. Once the Client query the Target by hostname, the Client will find out there is no proxy IP now. Therefore, in resolve library, the Client will find out the existing IP tunnel is not needed anymore, and will issue commands to clean up the IP tunnels, restore the routing table and firewall rules, with similar step ins (1 - 5).

In this schema, only the proxy servers are exposed to the clients. The proxy servers are equipped with DDoS IDS, like Snort, and function as the frontline fighting against possible DDoS attacks from malicious clients. We don't want to expose the Gateway or the Coordinator to the Client. Therefore the malicious clients can not find the Gateway or the Coordinator, and launch DDoS attacks against them.

In this schema, the indirect route needs to be set up at run time. Pre-setup tunnels might not fit the needs of the client because the chosen proxy server and the chosen gateway are not available until run time.

Advantage and disadvantage of this schema are as follows:

1) Once IP tunnel is set up, all the upper layer applications and protocols, like http, ftp, ICMP can use IP tunnel transparently.
2) The number of IP tunnels that we can set up is only limited by system resources on server.
3) There is overhead associated with IP tunnel. Usually the response time increase by 100% - 300%. But compared with the impact of DDoS attack, the overhead is still acceptable.
4) The client side needs to use the modified resolve library, which can be easily distributed through system update. The drawback is that the resolve library will only be called when client queries target by hostname. If client queries target by IP address, then the resolve library won't be called.
5) Needs to update DNS server and Bind software to support ALT type address.
6) Needs a set of participating proxy servers. If one proxy down, we can use other available proxy servers to enable indirect route.
7) The control messages communicated between SCOLD Daemon on the Proxy, Gateway and Target are SSL encrypted and mutually authenticated. There is overhead associated with that.

One interesting problem in this schema is the coordination between the Client and Proxy. After the Client sends control message to the Proxy and get confirmation message back, the Client will set up an IP tunnel towards the Proxy. At same time, the Proxy will also set up IP tunnel towards the Client. If for some reason, the Proxy fails to set up the IP tunnel, then the two-way communication between the Client and the Proxy is broken. Currently we use timeout on client side to solve this problem. If time out, the Client will pick another proxy severs and set up indirect route.

4.2. Indirect Route with Client Running SCOLD Daemon

The other way to implement indirect route of IP tunnel is to let a daemon server process running on the Client, Proxy, Gateway, Target and Coordinator machine, listening to a certain port, waiting for control message, setting up IP tunnels and setting routing tables / firewall rules accordingly. The control message communicated between SCOLD Daemon on the Coordinator, Proxy, Gateway and Target is SSL encrypted and mutually authenticated.

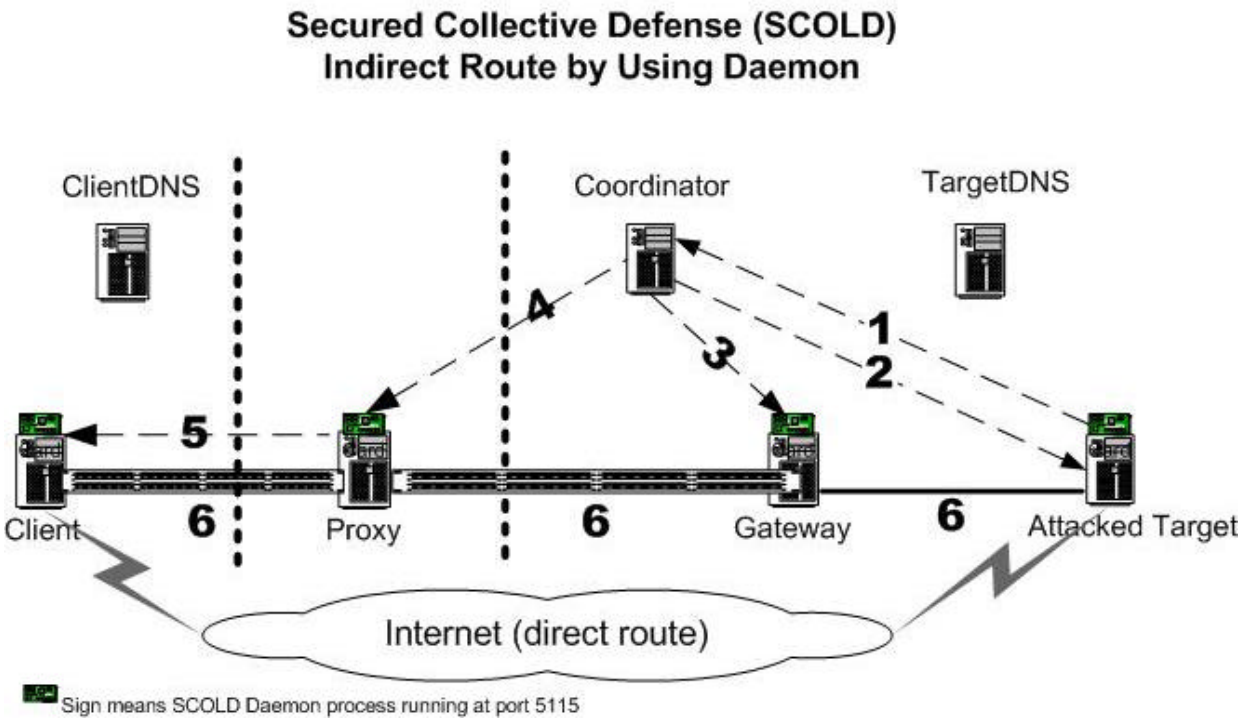

Figure 5: SCOLD Indirect Route by Using Daemon

The steps of setting up indirect route using daemons are as follows (Figure 5):

1. The Target gets DDoS attacks, raises alarm and notifies the Coordinator.
2. The Coordinator sends message to the Target, asking the Target to re-set its routing table and firewall rules.
3. The Coordinator sends message to the Gateway, asking it to re-set its routing table and firewall rules, and set up an IP tunnel to proxy.
4. The Coordinator sends message to the Proxy, asking it to re-set its routing table and firewall rules, set up IP tunnels to the Gateway and to the Client.
5. The Proxy sends message to the Client, asking it to re-set its routing table and firewall rules, set up an IP tunnel to the Proxy. Upon getting the control message from the Proxy, the Client needs to verify the Proxy, and set up IP tunnels to the Proxy accordingly.
6. The indirect route is set up. The Client can access the Target through indirect route.
7. Clean up the indirect route: When the DDoS attacks stop, the Target will notify the Coordinator. The Coordinator will issue commands to clean up the IP tunnels, restore the routing table and firewall rules, with similar steps in (1 - 6).

When the Proxy sends message to the Client, the Proxy needs to present a certificate signed by certain root CA, like Verisign, or the Coordinator. The Client needs to verify the certificate and decide whether it is acceptable. The Client may also query the ClientDNS, get the proxy IP addresses, and verify the IP of the Proxy.

Compared with schema in 4.2, this schema requires a SCOLD daemon running on client side. This may give the coordinator or the proxy server more control over the client, but it might also raise security concerns on client. But in this schema, the clients can get the IP addresses of proxy servers directly from the proxy servers, secure DNS update is necessary.

4.3. Indirect Route by Using Proxy as NAT Server

In this schema, the proxy server will function like a NAT server. The DNS Records in the clientDNS contain pairs of IP Address / Hostname. The Hostname is still the target name, but the IP address will be the proxy IP instead of real target IP. Therefore the Client's traffic towards the Target will be route to the Proxy. The Proxy will do a NAT translation, set up indirect route, and forward to the Target through indirect route.

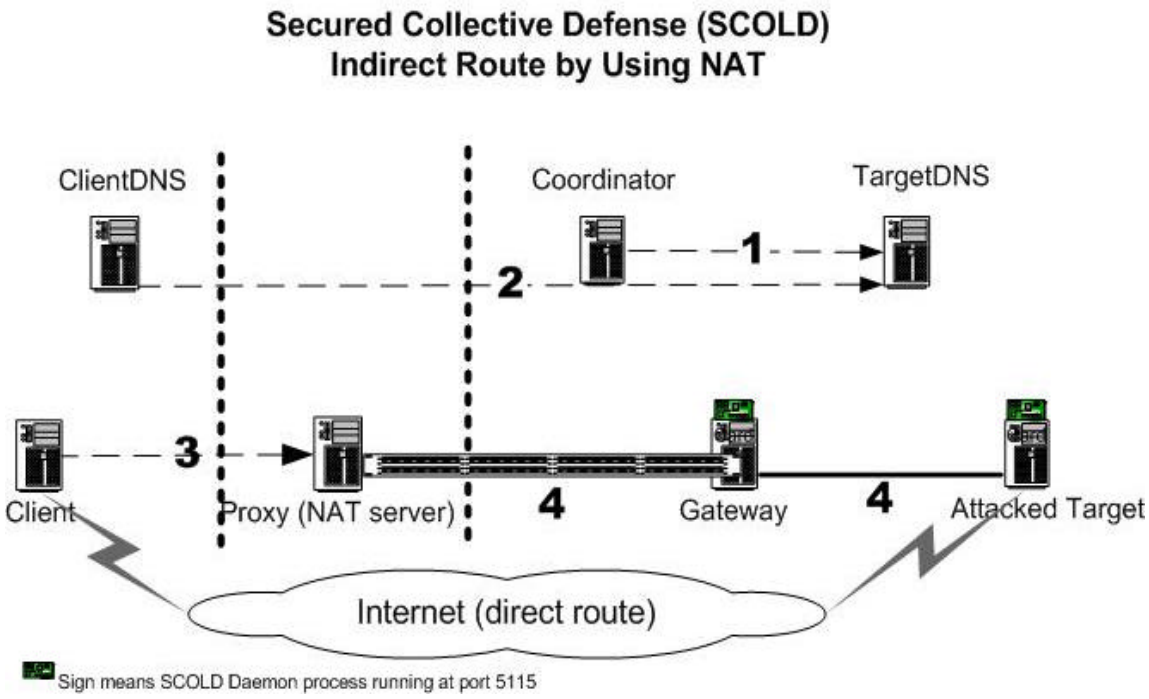

Figure 6: SCOLD Indirect Route by Using NAT

The steps are as follows (Figure 6):

1. The Coordinator notify the TargetDNS to update its DNS record by replacing the Target IP with the Proxy IP.
2. The Client queries the Target. The ClientDNS get updated DNS record from the TargetDNS.
3. The Client's traffic will be route to the Proxy.
4. The Proxy do a NAT translate, set up indirect route to the Target, and forward the client's traffic to the Target.

This schema requires no change on client side, and very little changes on ClientDNS to enable dynamic DNS update. But this schema has a scalable problem with large number of protected targets. Because it requires the proxy server to reserve a reasonable number of IP addresses, one to one corresponding to the Target. Also, the client will not get the real IP of target, and this might cause certain problems with some applications, like ssh, who cache the remote IP locally. Or it might raise a false alarm for IP spoofing if client runs certain internet security protection software or firewall.

4.4. Reset Indirect Route under severe DDoS attacks

The IP addresses of alternate gateways and coordinator are hidden from the clients, but the IP addresses of proxy servers are exposed to the clients. Therefore, the proxy servers are subject to the possible DDoS attacks.

The proxy servers are equipped with Intrusion Detection System (IDS). But under severe DDoS attacks, the selected proxy server might response slowly, or even stop responding. In this case, the indirect route needs to be reset up and old indirect route needs to be cleaned up.

The steps of re-setting up indirect route are as follows (Figure 7):

1. There is an indirect route set up already, but the selected Proxy is under severe DDoS attacks.
2. The attacked Proxy server will notify the Coordinator to re-set up the indirect route.
3. The Coordinator will notify the ClientDNS and the TargetDNS with secure DNS update, eliminating the Proxy from the list of available proxy servers.
4. The Coordinator will notify the Proxy, the Gateway, the Target to clean up the old indirect route.
5. The Client will temporarily lose connection to the Target. When time out, the Client will query the ClientDNS and get the new proxy server IP address.
6. The Client will set up indirect route though the newly selected proxy server.

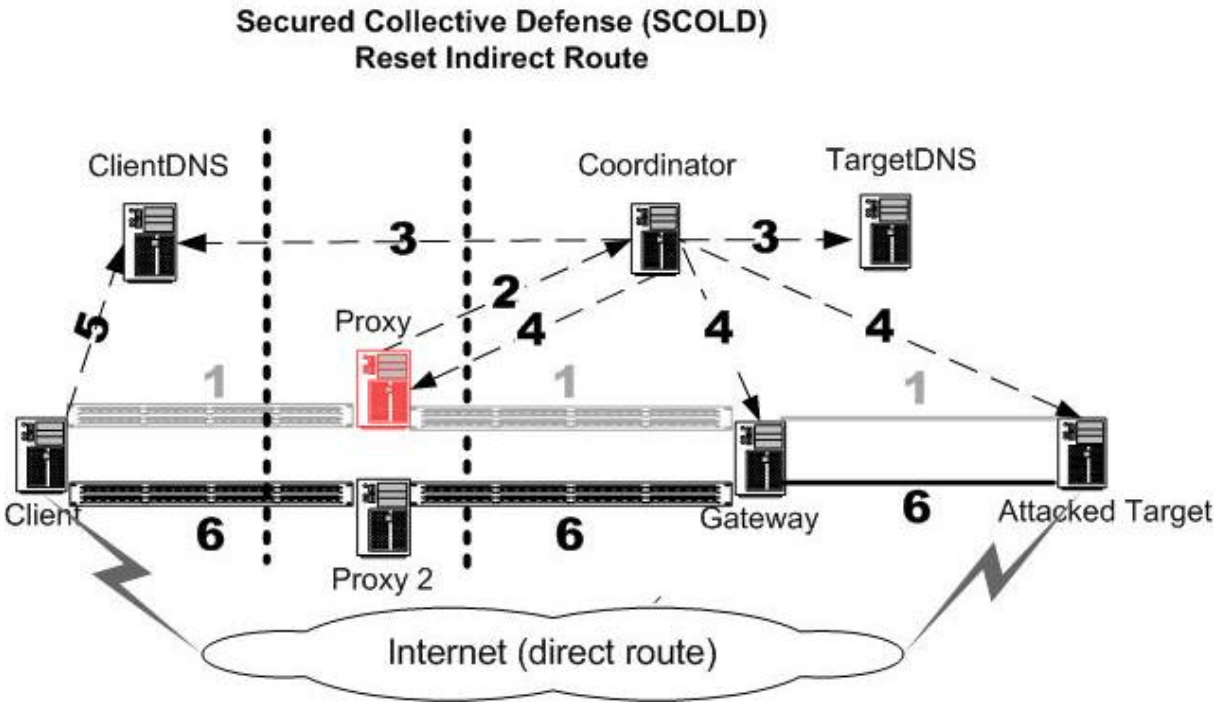

Figure 7: SCOLD Reset Indirect Route under DDoS attack

5. Test and Result.

To evaluate the overhead of indirect routing over IP tunnel and secure DNS update, we measure the response time and the throughput on direct and indirect routes in the testbed (Figure 8).

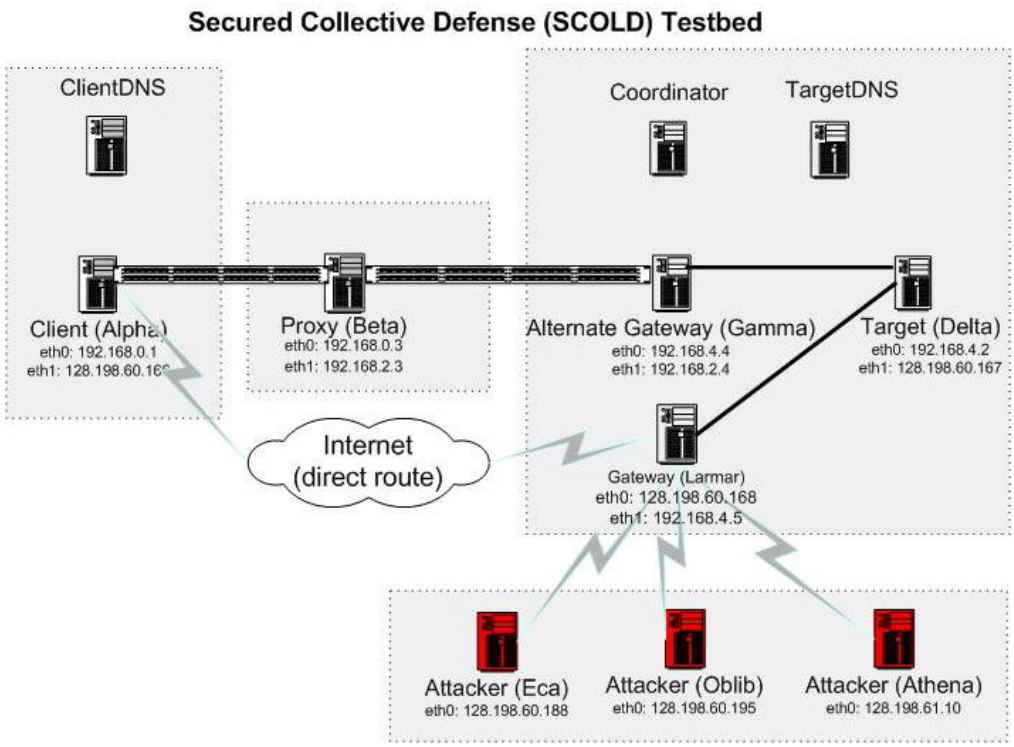

Figure 8: SCOLD testbed

5.1 Testbed machine configuration

Machine	Settings
Client	• Model: HP Vectra VL • CPU: Intel Pentium III 501.143 MHz • RAM: 255 MB • Hard Drive: 8455 MB • Network Interface: 3com 100 Mb • OS: Red Hat Linux 9.0
Proxy	Same
Gateway	Same
Target	Same
Coordinator	Same
ClientDNS	Same
TargetDNS	Same
Attacker	Same
Attacker	Same
Attacker	Same

StacheldrahtV4 [14] is used as the DDoS attacking tools. Among DDoS tools, StacheldrahtV4 is considered stable and sophisticated and is able to launch attacks in ICMP, UDP and TCP protocols.

5.2 Test Result and Analysis

Table 1: Ping Response Time Test (ping from client to target)

Test	No DDoS attack, direct route	DDoS attack, direct route	No DDoS attack, indirect route	DDoS attack, indirect route
Ping	0.49 ms	225 ms	0.65 ms	0.65 ms

In table 1, under DDoS attack, the ping response time increases dramatically. There are overhead associated with indirect route, but compared with impact of DDoS, the overhead is still acceptable. The response time of indirect route is not impacted too much by DDoS attack.

Table 2: SCOLD FTP/HTTP download Test (from client to target)

	No DDoS attack, direct		DDoS attack, direct route		No DDoS attack, indirect		DDoS attack, indirect	
Doc Size	FTP	HTTP	FTP	HTTP	FTP	HTTP	FTP	HTTP
100k	0.11 s	3.8 s	8.6 s	9.1 s	0.14 s	4.6 s	0.14 s	4.6 s
250k	0.28 s	11.3 s	19.5 s	13.3 s	0.31 s	11.6 s	0.31 s	11.6 s
500k	0.65 s	30.8 s	39 s	59 s	0.66 s	31.1 s	0.67 s	31.1 s
1000k	1.16 s	62.5 s	86 s	106 s	1.15 s	59 s	1.15 s	59 s
2000k	2.34 s	121 s	167 s	232 s	2.34 s	122 s	2.34 s	123 s

In table 2, under DDoS attack, the http/ftp download time increases dramatically. But the response time of indirect route is not impacted too much by DDoS attack.

Table 3: SCOLD Resolve Library Overhead Test

	Original Resolve Library	Enhanced Resolve Library
Ping	0.13 s	0.14 s
FTP download 100k	0.14 s	0.14 s
FTP download 500k	0.65 s	0.66 s
HTTP download 100k	4.5s	4.6 s
HTTP download 500k	31.1 s	31.1 s

In table 3, the overhead of modified resolve library is very limited.

Table 4: Time to Set up Indirect Route in SCOLD

Ping	Less than 1 s
FTP download	Less than 1 s
HTTP download	Less than 1 s

In table 4, the time used to set up indirect route is acceptable, compared with the possible delay caused by DDoS attack. The delay is primarily caused by the SSL authentication and communication.

As we can see from the above testing data, there is overhead associated with IP tunnel. The overhead occurs in the indirect route include more hop, more protocol processing (it goes through proxy server; and IP over IP overhead related to fragmentation and reassembly). But when the main gateway got attacked, the performance on the direct route will go from seconds to days or infinity. Therefore the IP tunnel overhead is still acceptable.

6. Conclusion and Future Work

It is our hope that the preliminary research results of the SCOLD project will produce a valuable secure software package, and provide valuable insights for the network security related proposals.

Currently we are focus on the secure DNS update and indirect route. Future works include:

1) Incorporate the Intrusion Detection and Isolation Protocol (IDIP) and Service Location Protocol (SLP) in the SCOLD design. These will enhance the existing A2D2 architecture and make it more robust [19, 20].
2) The Linux-based proxy server needs to be integrated with the enhanced intrusion detection SNORT plug-in created in the Autonomous Anti-DDoS test bed project [4].
3) Algorithms need to be designed to choose the best proxy server or subset of servers from a given set of proxy servers, to enable maximum bandwidth usage.
4) The infrastructure of SCOLD system provides the possibility of using multiple-path routing to get better performance and overcome the overhead of indirect route and SSL connection. We need to design the protocol and algorithm for multiple-path routing using proxy servers in SCOLD project.
5) Social study of collective defense dynamic, interaction and politic among participating organizations.
6) Recruit interested organization to join the SCOLD project and contribute resources like proxy servers.
7) Test SCOLD on large scale Internet domain with large number of clients. Find out the scalability of proposed schemas.

7. References

1. Internetnews.com, "Massive DDoS Attack Hit DNS Root Servers", http://www.internetnews.com/ent-news/article.php/1486981

2. David Moore, Geoffrey M. Voelker and Stefan Savage, "Inferring Internet Denial-of-Service Activity 2001", http://www.cs.ucsd.edu/~savage/papers/UsenixSec01.pdf

3. The SANS Institute, "How To Eliminate The Ten Most Critical Internet Security Threats" http://www.sans.org/top20/top10.php, 2001

4. Angela Cearns, Master Thesis "Design of an Autonomous Anti-DDoS network (A2D2)", http://cs.uccs.edu/~chow/pub/master/acearns/doc/angThesis-final.pdf, 2002

5. Steven Cheung, Ph.D. thesis "An Intrusion Tolerance Approach for Protecting Network Infrastructures", http://citeseer.nj.nec.com/cheung99intrusion.html, 1999

6. Jelena Mirkovic, Janice Martin and Peter Reiher, "A Taxonomy of DDoS Attacks and DDoS Defense Mechanisms", http://www.lasr.cs.ucla.edu/ddos/ucla_tech_report_020018.pdf

7. Network Associates Labs and Boeing, "IDIP Architecture", http://zen.ece.ohiou.edu/~inbounds/DOCS/reldocs/IDIP_Architecture.doc, 2002.

8. SVRLOC working group, "Service Location Protocol (SLP) Project", http://www.srvloc.org/

9. Michael D. Bauer," Securing DNS and BIND", ACM Linux Journal Volume 2000 Issue 78es, October 2000

10. John Viega, Matt Messier & Pravir Chandra, "Network Security with OpenSSL", O'Reilly, 2002

11. Edward. Chow, "Security Related Research Projects at UCCS network research lab", http://cs.uccs.edu/~Echow/research/security/uccsSecurityResearch.ppt, 2002

12. OpenSSL, "OpenSSL", http://www.openssl.org

13. Eric Green, "Glibc Howto", http://www.imaxx.net/~thrytis/glibc/Glibc2-HOWTO.html

14. Astalavista Network Library Archive. http://www.astalavista.com/archive/index.asp?dir=ddos

15. "Zebedee Secure IP tunnel", http://www.winton.org.uk/zebedee/

16. "SOCKS proxy server", http://www.tldp.org/HOWTO/Firewall-HOWTO-11.html

17. "IPIP tunnel", http://www.europe.redhat.com/documentation/HOWTO/Net-HOWTO/x1284.php3

18. "DNS BIND 9", http://www.isc.org/products/BIND/

19. Network Associates Labs and Boeing, "IDIP Architecture", http://zen.ece.ohiou.edu/~inbounds/DOCS/reldocs/IDIP_Architecture.doc, 2002.

20. SVRLOC working group, "Service Location Protocol (SLP) Project", http://www.srvloc.org/

21. D. G. Andersen, H. Balakrishnan, M. F. Kaashoek, R. Morris, "Resilient Overlay Networks," In Proceedings of 18th ACM SOSP, October 2001.

22. Information Sciences Institute, "Dynabone," http://www.isi.edu/dynabone/

23. J. Yan, S. Early, R. Anderson, "The XenoService – A distributed defeat for distributed denial of service", In Proceedings of ISW 2000, October 2000.

24. Asta Networks, "Vantage System Overview," http://www.astanetworks.com/products/vantage/

25. BBN Technologies, "Applications that participate in their own defense," http://www.bbn.com/infosec/apod.html