IP Fast Hopping Protocol Design

Vladimir Krylov and Kirill Kravtsov
Nizhny Novgorod State Technical University
n.a. R.E. Alekseev
Outline:

- Why we need a new DDoS protection mechanism?
- What principles we should follow?
- Main idea
- Frequency hopping
- Simplest scenario of brute-force DDoS attack
- IP Fast Hopping architecture
- IP Fast Hopping algorithm
- Basic implementation
- IP Fast Hopping advantages
- IP Fast Hopping limitations
- Conclusion
Why we need a new DDoS protection mechanism?

- A Denial-of-Service (DoS) attack is characterized by an explicit attempt to prevent the legitimate use of a service.
- A Distributed Denial-of-Service (DDoS) attack deploys multiple attacking entities to attain this goal.

Average peak attack bandwidth (Gigabits per second):

source: Prolexic Quarterly Global DDoS Attack Reports
What principles we should follow?

1. Real world applicability
   - software-based solution
   - re-use of already widely used technologies

2. The solution must be designed to prevent misuse
   - robustness of already established connections
The main idea

IP Fast Hopping is based on dynamic pseudorandom calculation of valid server’s IP address for each packet of each client session
Frequency hopping

The method is similar to frequency hopping:

Receiver and transmitter are switching from one frequency to other frequency synchronously during an ongoing data transmission session.
Simplest scenario of brute-force DDoS attack

- **Legitimate client**
- **DNS server**
- **Internet**
- **Botnet**
- **Internet-service (victim)**

Botnet consists of a large number of bots (Bot #1, Bot #2, ..., Bot #M) that launch an attack against the Internet-service (victim) by flooding it with requests.
IP Fast Hopping architecture
IP Fast Hopping algorithm

1. **Client** requests the IP address of the **Authorization server** using **DNS lookup**.
2. **Connection establishing** between the Client and the IP Hopper Core.
3. **Client’s authorization** is sent to the **Authorization server**.
4. **IP address of IP Hopper Manager** is sent to the **Authorization server**.
5. Request on establishing of enhanced secured connection is sent to the **Router #k**.
6. **Initial IP address of Internet-server** (IP₀) is sent to the **Router #k**.
7. **Destination IP address** = f(timestamp, ID)
8. **Source IP address** = f(timestamp, ID)
9. **IP pool, session UID** is sent to the **Router #m**.
10. **Connection established** between the **Router #k** and **Router #m**.
11. Request on switching to enhanced secured session is sent to the **IP Hopper Manager**.
12. **Destination IP = Server’s IP** is sent to the **IP Hopper Manager**.
13. **Source IP = Server’s IP** is sent to the **IP Hopper Manager**.
14. **Destination IP = Server’s IP** is sent to the **Internet-server**.
15. **Source IP = Server’s IP** is sent to the **Internet-server**.
Basic implementation

IP Hopper Core is new module of Linux build-in firewall Netfilter.

![Graph showing relative number of incoming packets for the victim server with and without IP Fast Hopping. The x-axis represents the number of malicious data streams, and the y-axis represents the relative number of incoming packets. The graph compares two categories: without IP Fast Hopping (diamonds) and with IP Fast Hopping (squares).]
IP Fast Hopping advantages

- Software solution utilizes existing network protocols
- Resistance to traffic interception
- Server’s protection against unauthorized access
- Hidden data destination
IP Fast Hopping limitations

- This particular implementation is limited to protect against TCP-based attacks
- Unprotected Authorization server
- Requires clients authorization
Conclusion

- We presented the new approach to prevent brute-force DDoS attacks
- The same method can be used to hide content and destination of communication session
Десятая независимая научно-практическая конференция «Разработка ПО 2014»

23 - 25 октября, Москва

Thank you

vkrylov@heterarchica.com
kirill@kravtsov.biz