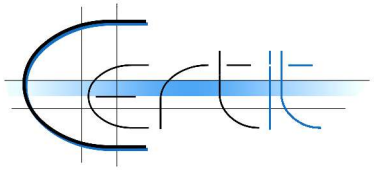
NAL
Network Auditing Language

Lorenzo Martignoni, Stefano Masiero



NAL
Network Auditing Language

- Linguaggio di scripting per interagire in modo semplice con la rete
- Supporta la configurazione, l'invio e la ricezione di singoli pacchetti (*packet*)
- Supporta la configurazione e la realizzazione di canali di comunicazione (*stream* TCP o UDP)

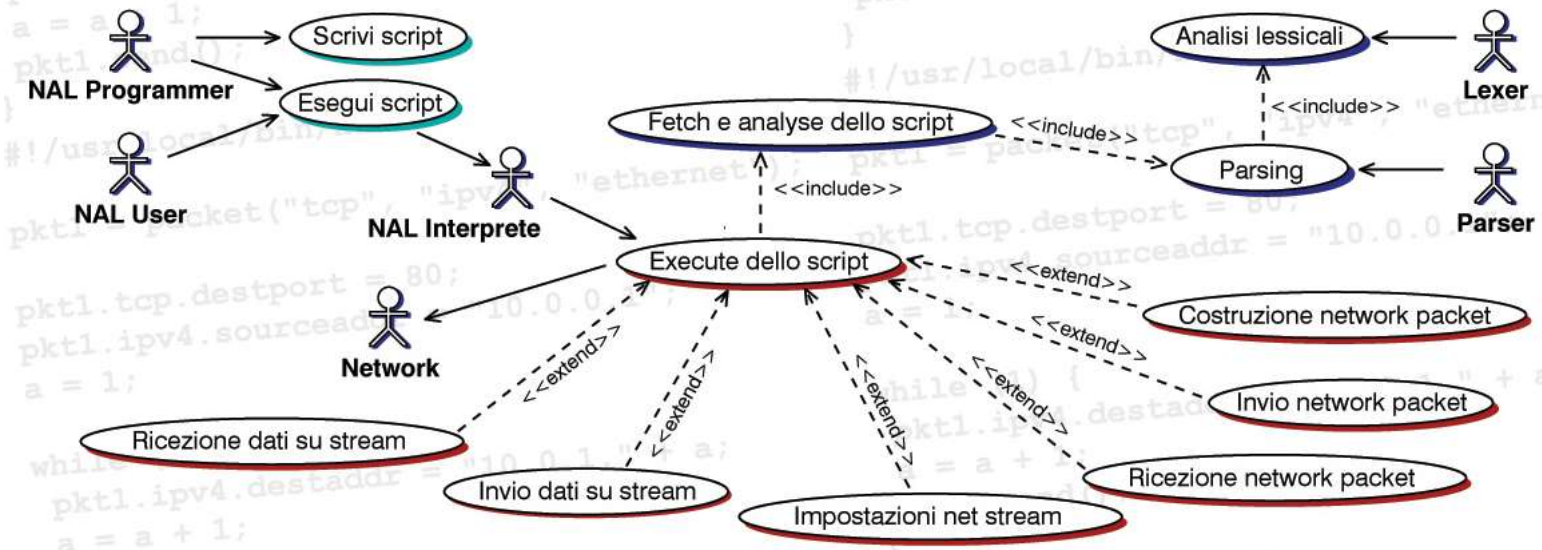
Obiettivi

- Nascondere la complessità e la ripetitività della programmazione di rete a basso livello
- Adattare velocemente le applicazioni esistenti per fronteggiare nuove improvvise esigenze

Gestione della complessità

- I costrutti messi a disposizione dal linguaggio sono pensati per semplificare la gestione delle strutture dati di rete e delle operazioni sulle stesse
- Il linguaggio di scripting permette di “giocare” inventando nuove applicazioni concentrandosi solo sullo scopo

Scenario


NAL Programmer
Scrivi script
Esegui script
NAL User
NAL Interprete
Network
Fetch e analyse dello script
<<include>>
Execute dello script
Analisi lessicali
Lexer
<<include>>
Parsing
Parser
<<extend>>
Costruzione network packet
<<extend>>
Invio network packet
<<extend>>
Ricezione network packet
<<extend>>
Impostazioni net stream
<<extend>>
Invio dati su stream
<<extend>>
Ricezione dati su stream

Funzioni principali

- Costruzione e manipolazione di singoli pacchetti, lavorando direttamente sui campi delle intestazioni di ogni livello dello stack ISO/OSI
- Invio e ricezione di singoli pacchetti
- Creazione e configurazione di canali di comunicazione ad alto livello (connessioni TCP e UDP)

Caratteristiche principali

- Due strutture dati principali
  - *packet*
  - *stream*
- Estendibilità delle funzioni del linguaggio
- Estendibilità dei protocolli dello stack ISO/OSI supportati

Caratteristiche secondarie

- Gestione delle strutture complesse come “oggetti”
- Tipizzato dinamicamente
- Non dichiarativo

Esempio

```
#!/usr/local/bin/nal
pkt1 = packet("tcp", "ipv4");
pkt1.tcp.destport = 80;
pkt1.ipv4.sourceaddr = "10.0.0.1";
a = 1;
while (a <= 255) {
        pkt1.ipv4.destaddr = "10.0.1." + a;
        a = a + 1;
        pkt1.send();
}
```

Esempio

```
#!/usr/local/bin/nal
include("portscan.nal");
i = 1;
while (i < 255) {
   open_ports = portscan("192.168.132." + i);
   k = 0;
   while (k < open_ports.length) {
      str = "host 192.168.132.“ + i + “ “ + open_ports[k]);
      print(str, “\n”);
      k = k + 1;
   }
   i = i + 1;
}
```

Applicabilità

- Testing della configurazione di apparti e servizi di rete
- Rilevazione di vulnerabilità note (worm, backdoor, …)
- Ricerca di nuove vulnerabilità
- Monitoraggio di servizi
- Supporto alla didattica

Sito ufficiale

http://nal.sourceforce.net