

USO DE LOTOS NA GERÊNCIA DE SEGURANÇA DE APLICAÇÕES DISTRIBUÍDAS EM JAVA

ALEXANDRE VELOSO DE MATOS
alex@lrg.ufsc.br

BERNARDO GONÇALVES RISO
riso@lrg.ufsc.br

MIRELA SECHI MORETTI ANNONI NOTARE
mirela@lrg.ufsc.br



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CURSO DE PÓS-GRADUAÇÃO EM CIÊNCIAS DA COMPUTAÇÃO
DEPARTAMENTO DE INFORMÁTICA E DE ESTATÍSTICA
LABORATÓRIO DE REDES E GERÊNCIA www.lrg.ufsc.br
CAMPUS UNIVERSITÁRIO DA TRINDADE – CAIXA POSTAL 476
TELEFONE: (048) 331-9738 - FAX: (048) 331-9770
88040-900 FLORIANÓPOLIS – SC

RESUMO

Apesar de possuir características que a tornam mais flexível que outras linguagens de programação, existem fragilidades que podem facilitar a construção de ataques usando a linguagem Java.

Um componente de extrema importância, no Modelo de Segurança da plataforma Java é o Carregador de *Applets*, responsável, entre outras ações, pela associação dinâmica de uma classe remota (incluída em alguma *applet*) e seu respectivo código. A adição de classes a um ambiente de execução Java pode ser danosa quando se tratar, por exemplo, de uma *applet* com intenções maliciosas. Neste caso, a possibilidade de existirem classes com métodos perigosos que podem, por exemplo, sobrepor-se à definição original de classes predefinidas, pode ser impedida pelo trabalho do Carregador de Classes (CCA).

Neste trabalho, a Técnica de Descrição Formal LOTOS e a ferramenta Eucalyptus são usadas com a intenção de testar e validar a implementação de uma extensão à especificação original do Carregador de Applets. Abaixo é apresentada a especificação da proposta utilizando-se LOTOS.

```
specification CCAEx [remot, load, local, name] : noexit  
behaviour  
  hide names in Carga [load, remot, names, local]  
  |[names]|  
  Scache [names, map, name]  
  where  
  process Carga [load, remot, names, local]:noexit:=  
  load; names; (local; Carga [load, remot, names, local]  
    [] i; remot; Carga [load, remot, names, local])  
  endproc  
  
  process Scache [names, map, name]:noexit:=  
  names; map;(Scache [names, map, name]  
    []i; name; stop)  
  endproc  
endspec
```

REFERÊNCIAS BIBLIOGRÁFICAS

- [01] E. Brinskma. “A tutorial of LOTOS, document IS 8807”.
- [02] G. McGraw e E. Felten. “Java™ Security. *Hostile Applets, Holes, and Antidotes*”. Wiley Computer Publishing. New York. 1997.
- [03] D. Wallach, D. Balfanz, D. Dean and E. Felten. “Extensible Security Architectures for Java”. <http://www.cs.princeton.edu/sip/sosp97.html>. in 16th Symposium on Operating Systems Principles. Saint-Malo. 1997. pp 116-128.