

Formalisation et génération d'injections

LAAS-CNRS – INSA Toulouse

Eric Alata	eric.alata@laas.fr
Mohamed Kâaniche	mohamed.kaaniche@laas.fr
Vincent Nicomette	vincent.nicomette@laas.fr
Didier Le Botlan	didier.le.botlan@laas.fr

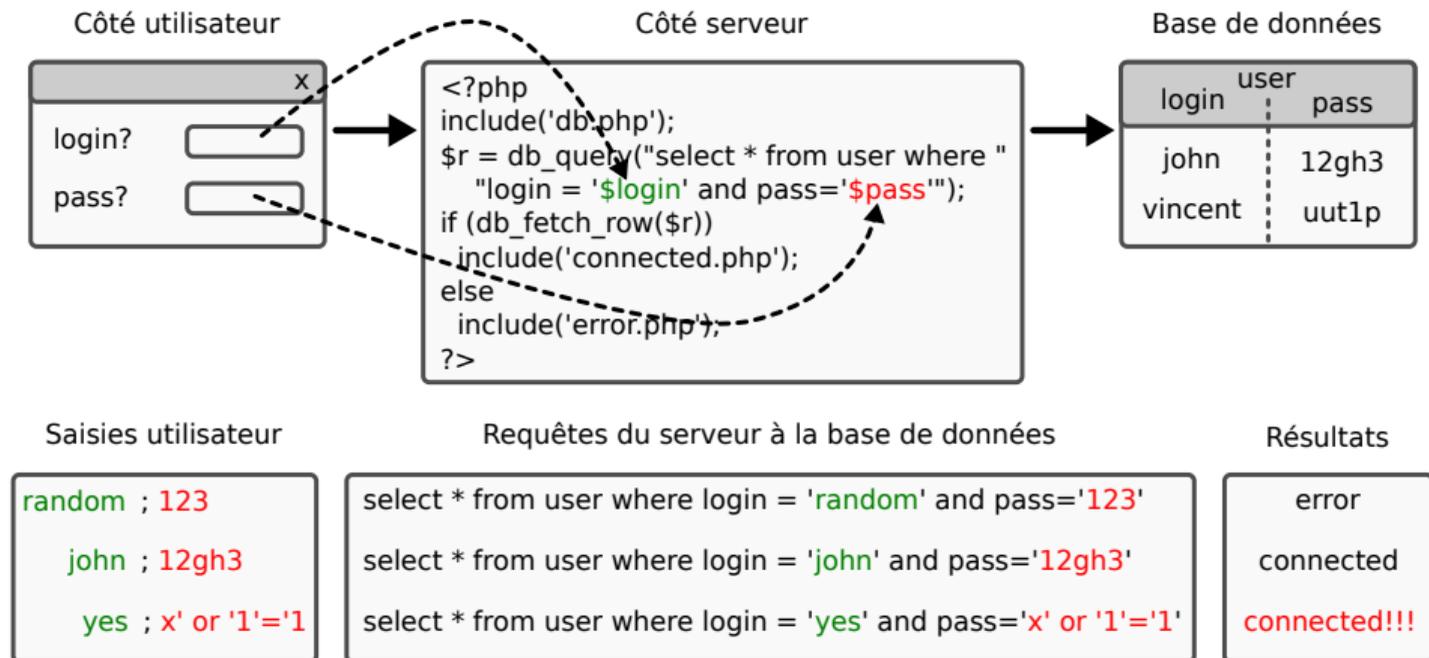
Mai 31, 2018

Sécurité au LAAS – et langages

- ▶ Fuzzer pour applications web
- ▶ IDS embarqué pour l'automobile – thèse de Ivan Studnia
- ▶ **Étude formelle des injections**
- ▶ Obfuscation de protocoles de communication – thèse de Julien Duchene

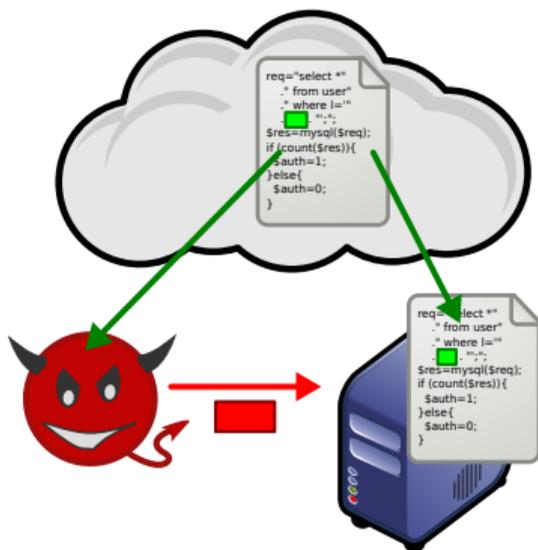
Étude formelle des injections

Rappel sur les injections SQL

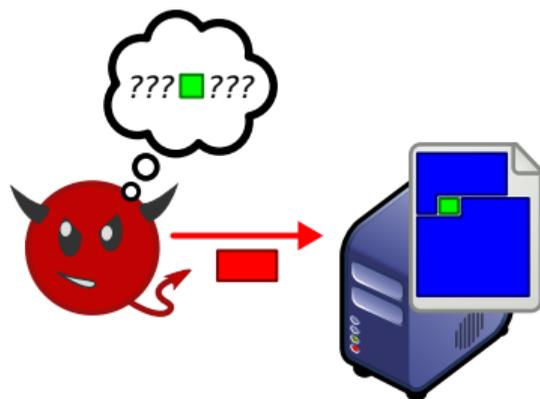


Le client choisit les données qu'il envoie de manière à ce que la sémantique de la requête envoyée au serveur de base de données soit changée

Stratégies pour la découverte des injections



Stratégie boîte blanche



Stratégie boîte noire

Problématique et approches

Le développeur ou l'expert en sécurité souhaite...

- ▶ savoir si le langage qu'il manipule est “vulnérable” aux injections
- ▶ savoir si la requête qu'il construit est “vulnérable” aux injections
- ▶ disposer d'un outil de test des injections efficace

Approches disponibles dans la littérature

- ▶ Détection de fautes dans les logiciels
ex. : grammaires [Rud07], algorithmes génétiques [GHGT⁺ 08]
- ▶ Détection des vulnérabilités liées aux injections
ex. : aléatoire [MZR], injection sheet cheat
- ▶ Détection des attaques par injections
ex. : grammaires [KT08]

⇒ Quid de la détection des vulnérabilités basées sur les grammaires ?

Rappel sur la théorie des langages

$G = (T, N, R, S) \Rightarrow$ langage \Rightarrow phrases

T3 : grm. régulière

$S \rightarrow l$
 $S \rightarrow p$
 $S \rightarrow lN$
 $S \rightarrow pN$
 $N \rightarrow ,lN$
 $N \rightarrow ,pN$
 $N \rightarrow \&l$
 $N \rightarrow \&p$

$S \Rightarrow^* l, p, l\&p$

$[lp] \mid [lp](,[lp])^*\&[lp]$

T2 : grm. algébrique

$S \rightarrow \epsilon$
 $S \rightarrow ES$
 $E \rightarrow L$
 $E \rightarrow A$
 $L \rightarrow (S)$
 $A \rightarrow \text{token}$

$S \Rightarrow^*$ (if
(equal
root-pwd
"rpwd")
(set-auth
1)
(error
"bad pwd"))

T1 : grm. contextuelle

$S \rightarrow \epsilon$
 $S \rightarrow abc$
 $S \rightarrow aTBc$
 $T \rightarrow abC$
 $T \rightarrow aTBC$
 $CB \rightarrow BC$
 $bB \rightarrow bb$
 $Cc \rightarrow cc$

$S \Rightarrow^* aaabbbccc$

Langage $a^n b^n c^n$

Rappel sur la théorie des langages

Type	Nom	Forme des rgles
T0	Grammaire générale	aucune restriction
T1	Grammaire monotone	$\alpha \rightarrow \beta$ avec $ \alpha \leq \beta $
T1	Grammaire contextuelle	$\alpha A \beta \rightarrow \alpha \gamma \beta$ avec $A \in N$ et $ \gamma > 0$
T2	Grammaire algébrique	$A \rightarrow \beta$ avec $A \in N$
T3	Grammaire régulière	$A \rightarrow \beta B$ avec $A \in N, B \in N, \beta \in T^*$

Classification de Chomsky [Cho56]

Opération	T0	T1	T2	T3
$\mathcal{L}(G_i^a) \cap \mathcal{L}(G_j^b)$	✓	✓	✗	✓
$\mathcal{L}(G_3^a) \cap \mathcal{L}(G_j^b)$	✓	✓	✓	✓
$\mathcal{L}(G_3^a) \setminus \mathcal{L}(G_j^b)$	✓	✗	✓	✓
homomorphisme	✓	✗	✓	✓
homomorphisme sans effacement	✓	✓	✓	✓

Propriétés de clôture [HU69] (G_i, G_i^a et G_j^b sont des grammaires de type i)

Problème	T0	T1	T2	T3
$\mathcal{L}(G_i) = \emptyset ?$	I	I	D	D
$\mathcal{L}(G_i^a) \cap \mathcal{L}(G_j^b) = \emptyset ?$	I	I	I	D
Pour $\alpha \in T^*, \alpha \in \mathcal{L}(G_i) ?$	I	D	D	D

Problèmes décidables [Lan64] (D : problème décidable ; I : problème indécidable)

Formalisation des injections

Définition (Point d'injection)

Un point d'injection est un *token* dont la valeur est renseignée par l'utilisateur

Définition (Injections sur un langage)

Étant donné un langage \mathcal{L} et un point d'injection x , on définit les *injections de \mathcal{L} en x* , noté $I_x(\mathcal{L})$, comme étant le langage défini par $I_x(\mathcal{L}) = \{w \mid \exists p \in T^*, s \in T^* \text{ t.q. } pxs \in \mathcal{L} \wedge pws \in \mathcal{L}\}$

Définition (Injections sur une phrase d'un langage)

Étant donné un langage \mathcal{L} , une phrase pxs et un point d'injection x , on définit les *injections de \mathcal{L} en x pour la phrase (p, x, s)* , noté $I_x^p(\mathcal{L}, p, s)$, comme étant le langage défini par $I_x^p(\mathcal{L}, p, s) = \{w \mid pxs \in \mathcal{L} \wedge pws \in \mathcal{L}\}$

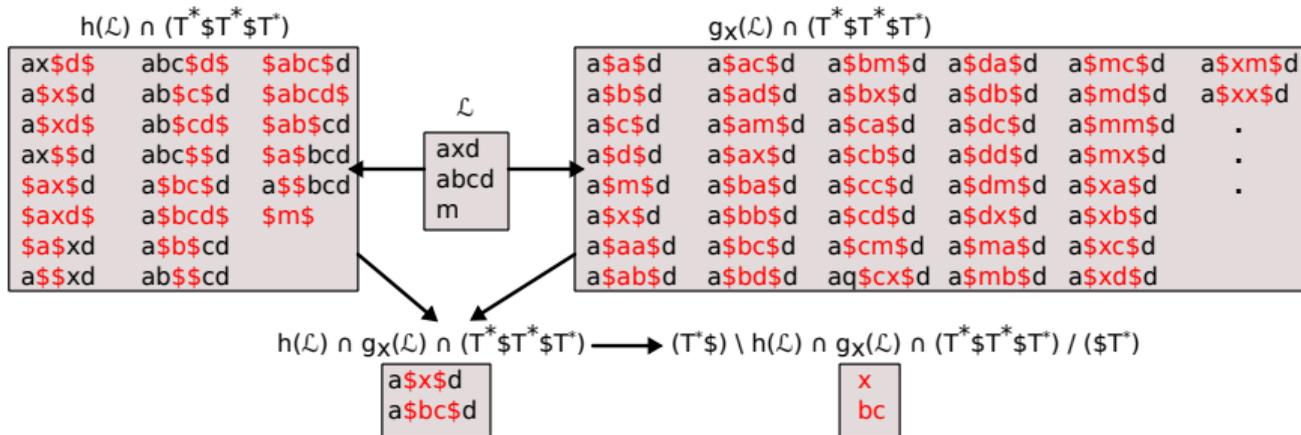
Définition (Langage vulnérable)

Un langage \mathcal{L} est vulnérable vis-à-vis d'un point d'injection x si $\exists w \in I_x(\mathcal{L})$ t.q. $|w| > 1$

Injections sur un langage régulier

Peut-on construire une grammaire des injections
pour un langage régulier ? **oui !**

- ▶ Soient \mathcal{L} un langage régulier et $\$ \notin T$ une nouvelle lettre
- ▶ Homomorphisme $h \left\{ \begin{array}{l} h(a) = \{a, \$a, a\$, \$a\$\} \quad \forall a \in T \end{array} \right.$
- ▶ Homomorphisme $g_x \left\{ \begin{array}{l} g_x(x) = \$T*\$ \\ g_x(a) = \{a\} \quad \forall a \in (T - \{x\}) \end{array} \right.$



Injections sur un langage régulier

Peut-on construire une grammaire des injections
pour un langage régulier ? **oui !**

- ▶ Soient \mathcal{L} un langage régulier et $\$ \notin T$ une nouvelle lettre
- ▶ Homomorphisme $h \begin{cases} h(a) = \{a, \$a, a\$, \$a\$\} \quad \forall a \in T \end{cases}$
- ▶ Homomorphisme $g_x \begin{cases} g_x(x) = \$T^*\$ \\ g_x(a) = \{a\} \quad \forall a \in (T - \{x\}) \end{cases}$

$l, p, \blacksquare \& p$	$l, p, l \& \blacksquare$
l	l
p	p
l, p	
...	

$l, p, \blacksquare \& p$ est une phrase vulnérable : $[lp](, [lp])^*$

$l, p, l \& \blacksquare$ est une phrase non vulnérable

Injections sur un langage algébrique

Étant donné un langage algébrique et un point d'injection sur ce langage, peut-on construire une grammaire algébrique qui engendre les injections associées ? **non !**

- ▶ L'opération d'injection ne préserve pas l'algébricité des langages
- ▶ Exemple avec $a^{2^n} b^{2^n} c^{2^n} w$

S	\rightarrow	S_1	S_1	\rightarrow	v	T_1	v	x	S_2	\rightarrow	d	v	a	b	c
S	\rightarrow	S_2	T_1	\rightarrow	d				S_2	\rightarrow	a	S_2	m		
			T_1	\rightarrow	a	T_1	a		S_2	\rightarrow	b	S_2	n		
			T_1	\rightarrow	b	T_1	b		S_2	\rightarrow	c	S_2	o		
			T_1	\rightarrow	c	T_1	c		S_2	\rightarrow	m	S_2	a	a	
			T_1	\rightarrow	m	T_1	m		S_2	\rightarrow	n	S_2	b	b	
			T_1	\rightarrow	n	T_1	n		S_2	\rightarrow	o	S_2	c	c	
			T_1	\rightarrow	o	T_1	o		S_2	\rightarrow	v	S_2	w		
			T_1	\rightarrow	v	T_1	v		S_2	\rightarrow	w	S_2	v		
			T_1	\rightarrow	w	T_1	w								

Injections sur un langage algébrique

Pire ! Quelque soit le langage de type 0, il existe un langage algébrique et un point d'injection pour lesquels le langage des injections correspond au langage de type 0
(\Rightarrow caractérisation des langages de type 0)

- ▶ Intuition : “les langages algébriques ne sont pas clos avec l'opération de quotient par d'autres langages algébriques”
- ▶ Démonstration générale \Rightarrow basée sur les travaux de [LT90]

Every recursively enumerable language $\mathcal{L} \subseteq \Sigma^$ can be expressed in the form*

$$\mathcal{L} = \{h(\alpha) \setminus g(\alpha) \mid \alpha \in \mathcal{R}^+\} \cap \Sigma^*$$

\mathcal{R} is an ϵ -free regular set, h is a nonerasing homomorphism, and g is an homomorphism 3-limited on \mathcal{R}^+

Comment montrer qu'un langage n'est pas vulnérable ?

Injections sur une phrase d'un langage algébrique

Peut-on calculer la grammaire des injections associée à une phrase donnée d'un langage algébrique ? **oui !**

- ▶ Rappel : $I_x(\mathcal{L}) = \{w \mid \exists p \in T^*, s \in T^* \text{ t.q. } pxs \in \mathcal{L} \wedge pws \in \mathcal{L}\}$
- ▶ Langage contenant une seule phrase issue d'un langage algébrique \rightarrow langage régulier
- ▶ Les langages algébriques sont clos avec l'opération de quotient par un langage régulier

Injections sur une phrase d'un langage algébrique

$S \rightarrow \epsilon$
 $S \rightarrow E S$
 $E \rightarrow L$
 $E \rightarrow A$
 $L \rightarrow (S)$
 $A \rightarrow \text{token}$

$S \Rightarrow^*$ (if
▷ (equal
▷▷ root-pwd
▷▷ ■)
▷ (set-auth
▷▷ 1)
▷ (error
▷▷ "bad pwd"))

$X'10 \rightarrow S'3 E'0$
 $X'10 \rightarrow E'1 S'1$
 $X'10 \rightarrow E S$
 $X'10 \rightarrow (S)$
 $X'10 \rightarrow ()$
 $X'10 \rightarrow \text{token}$
 $X'10 \rightarrow$
 $X'1 \rightarrow E'3)$
 $X'1 \rightarrow)$
 $X'1 \rightarrow E'1 S$
 $E'0 \rightarrow E E'0$
 $E'0 \rightarrow (S'1$
 $E'1 \rightarrow E'3)$
 $E'1 \rightarrow)$

$E'3 \rightarrow (S)$
 $E'3 \rightarrow ()$
 $E'3 \rightarrow \text{token}$
 $E'3 \rightarrow E S$
 $S'1 \rightarrow E S'1$
 $S'1 \rightarrow (S$
 $S'1 \rightarrow ($
 $S'3 \rightarrow X'1)$
 $S \rightarrow E S$
 $S \rightarrow (S)$
 $S \rightarrow ()$
 $S \rightarrow \text{token}$
 $E \rightarrow (S)$
 $E \rightarrow ()$
 $E \rightarrow \text{token}$

Injections sur une phrase d'un langage algébrique

$S \rightarrow \epsilon$
 $S \rightarrow E S$
 $E \rightarrow L$
 $E \rightarrow A$
 $L \rightarrow (S)$
 $A \rightarrow \text{token}$

$S \Rightarrow^*$ (if
▷ (equal
▷▷ root-pwd
▷▷ ■)
▷ (set-auth
▷▷ 1)
▷ (error
▷▷ "bad pwd"))

$S \Rightarrow^*$ (if
▷ (equal
▷▷ root-pwd
▷▷ "abc")
▷ (print
▷▷ (get-pwd
▷▷▷ "root"))
▷ (print
▷▷ (get-pwd
▷▷▷ "root"))
(if
▷ (equal
▷▷ "str"
▷▷ "str")
▷ (set-auth
▷▷ 1)
▷ (error
▷▷ "bad pwd"))

Résultats

- ✓ Formalisme des injections
- ✓ Construction des injections sur un langage régulier
- ✓ Construction des injections pour une phrase d'un langage algébrique
- ✗ Complexité pour les langages algébriques

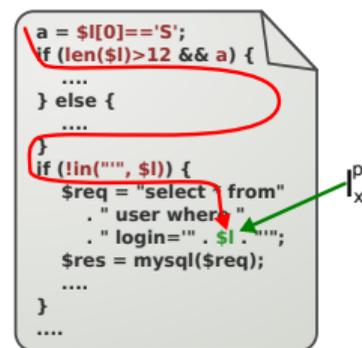
Perspectives

- ✓ Construction des injections pour une phrase d'un langage algébrique

$$I_x^P(\mathcal{L}, p, s) = \{w \mid pxs \in \mathcal{L} \wedge pws \in \mathcal{L}\}$$

$I_x^P(\mathcal{L}, p, s) \neq \emptyset \Rightarrow$ vulnérabilité ?

- ▶ Tests sur les données mis en place par le développeur
- ▶ Représentation de ces tests par une expression régulière *pas toujours possible...*
- ▶ Intersection entre cette expression régulière et $I_x^P(\mathcal{L}, p, s)$
- ▶ Langage vide \Rightarrow pas d'injections



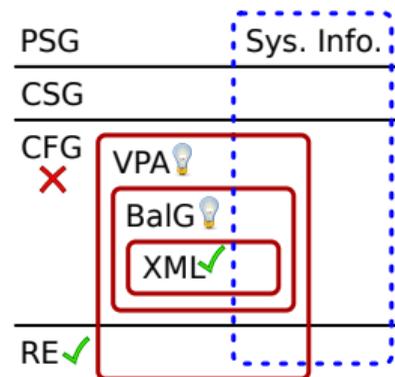
```
a = ${l[0]}=='S';
if (len($l)>12 && a) {
  ....
} else {
  ....
}
if (!in("", $l)) {
  $req = "select      from"
        . " user where"
        . " login='" . $l . "'";
  $res = mysql($req);
  ....
}
....
```

$(len($l)>12)\&(\$l[0]=='S')\&!in("", \$l) \mid$
 $(len($l)\leq 12)\&!in("", \$l) \mid \dots$

Perspectives

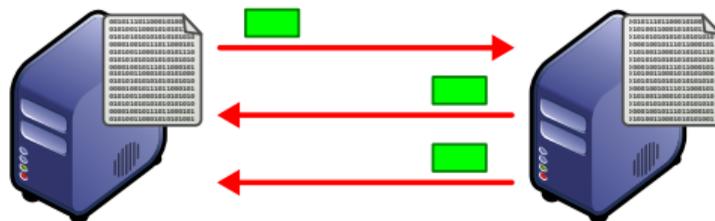
✗ Complexité pour les langages algébriques

- ▶ Simplification d'une grammaire en fonction du point d'injection
⇒ langage des injections algébrique ?
- ▶ Grammaires dont le langage des injections est algébrique ?
- ▶ Quid de certaines sous-familles des langages algébriques ?
- ▶ Approximation régulière d'une grammaire algébrique
⇒ approximation du langage des injections ?



Obfuscation de protocoles de communication

Protocoles de communication



- ▶ Assure la cohérence des échanges
- ▶ Deux parties : format des messages et structure des échanges
- ▶ Focus sur le format des messages

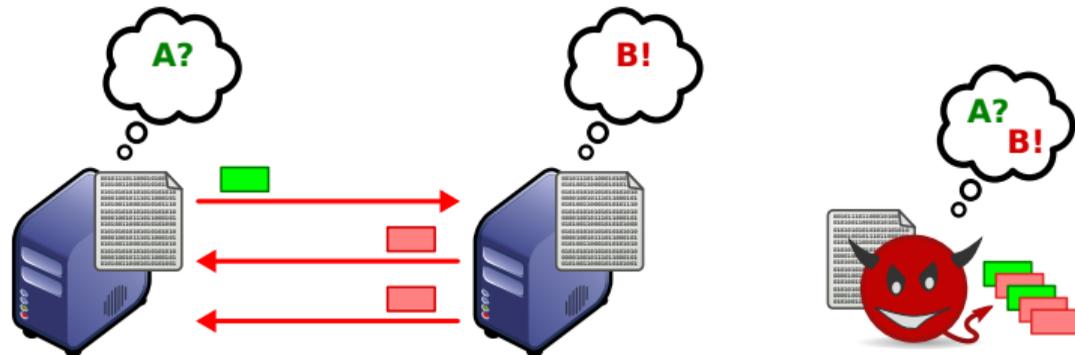
- ▶ Messages *typés*
- ▶ Analyse du contenu d'un message déterministe et rapide
- ▶ \Rightarrow machine à état \Rightarrow automate finis déterministe

Rétro-conception de protocoles de communication

Plusieurs outils automatiques disponibles

- ▶ Netzob, Discoverer, ReverX, ASAP, ScriptGen, ...
- ▶ Analyse d'une application binaire, d'une trace d'exécution ou d'une **trace réseau**
- ▶ Démarches similaires
 1. Classification des messages en fonction d'une métrique de similarité
 2. Pour chaque classe de messages : identification des délimiteurs et des champs de taille fixe
⇒ inférence du format des messages
 3. Analyse de la séquence des classes de messages
⇒ inférence du protocole d'échange
- ▶ Modèles utilisés pour représenter le format et le protocole : automates

Comment empêcher une troisième entité, qui a accès au binaire des applications et au réseau, de comprendre le contenu des échanges ?



Approches et proposition

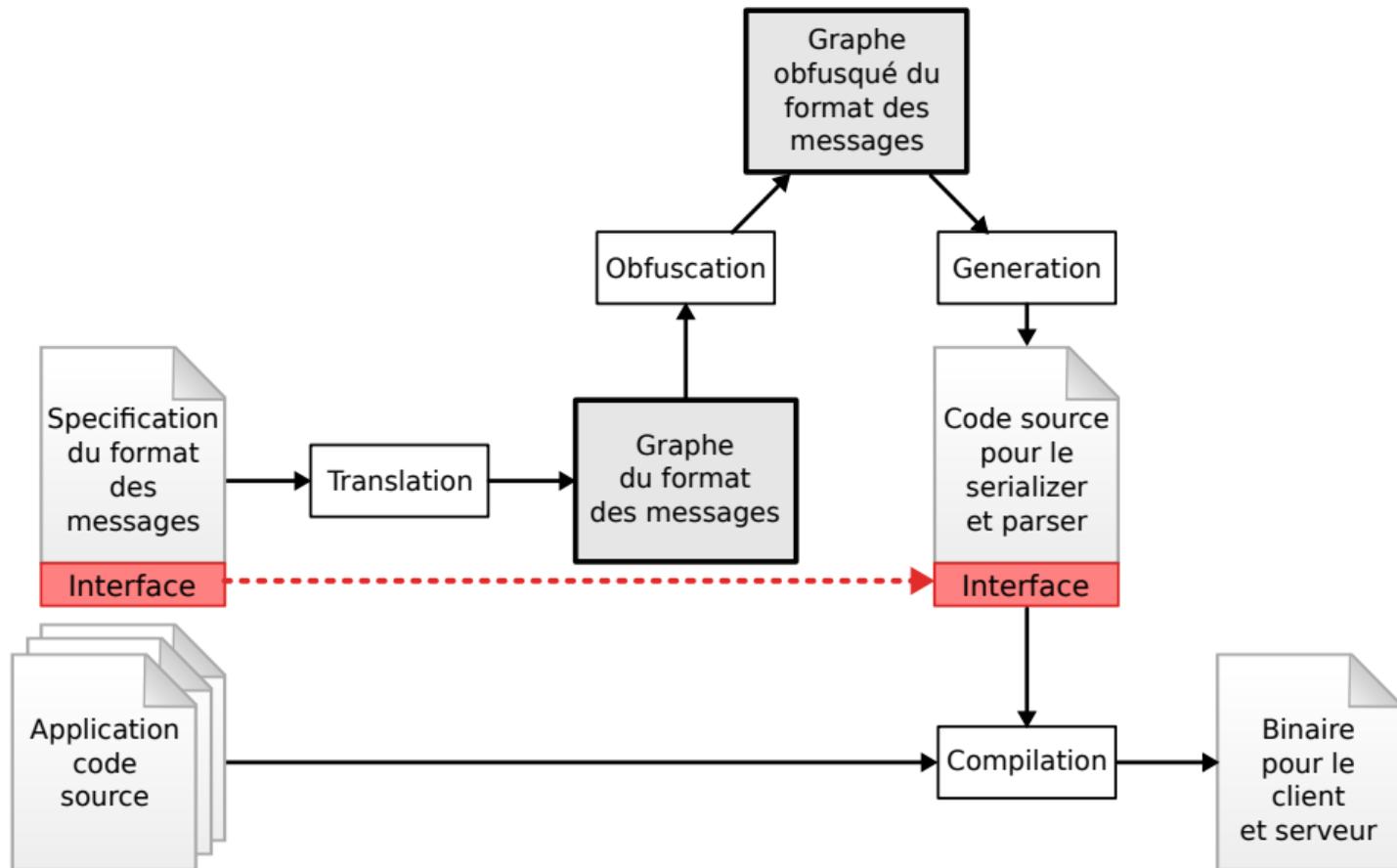
Approches habituelles \Rightarrow focus sur la confidentialité

- ▶ Cryptographie
- ▶ Obfuscation du binaire de l'application
- ▶ Obfuscation des messages au moment de leur envoi sur le canal

Proposition \Rightarrow contrer les outils automatiques de rétro-conception

- ▶ Obfuscation du format des messages directement
- ▶ Idée : complexifier le format tout en restant déterministe
i.e., profiter des limites des outils automatiques
- ▶ Grammaire régulière \rightarrow grammaire algébrique déterministe
- ▶ Intuition : $(k = v)^* \Rightarrow k^n (= v)^n$
- ▶ Représentation du format avec un graphe et transformations sur ce graphe

Implémentation



Transformations

SplitAdd	
<p>The diagram illustrates the SplitAdd transformation. On the left, a rounded rectangle contains a box labeled 'X' above the text 'Te/L(n)'. An arrow points to the right, where another rounded rectangle contains a box labeled 'X'' above 'S/Dgt'. From the bottom of the 'X'' node, two arrows point down to two separate rounded rectangles. Each of these contains a box labeled 'X1' and 'X2' respectively, with 'Te/L(n)' written below each box.</p>	Serialization pseudocode ↓
	Choose a random value $X1$ Compute $X2 = X + X1$
	Constraints
	<i>Boundary of parent nodes must be either Delegated or End</i>
Challenge	
Inference models and classification: more dependencies between fields in message and various representations of the same message	

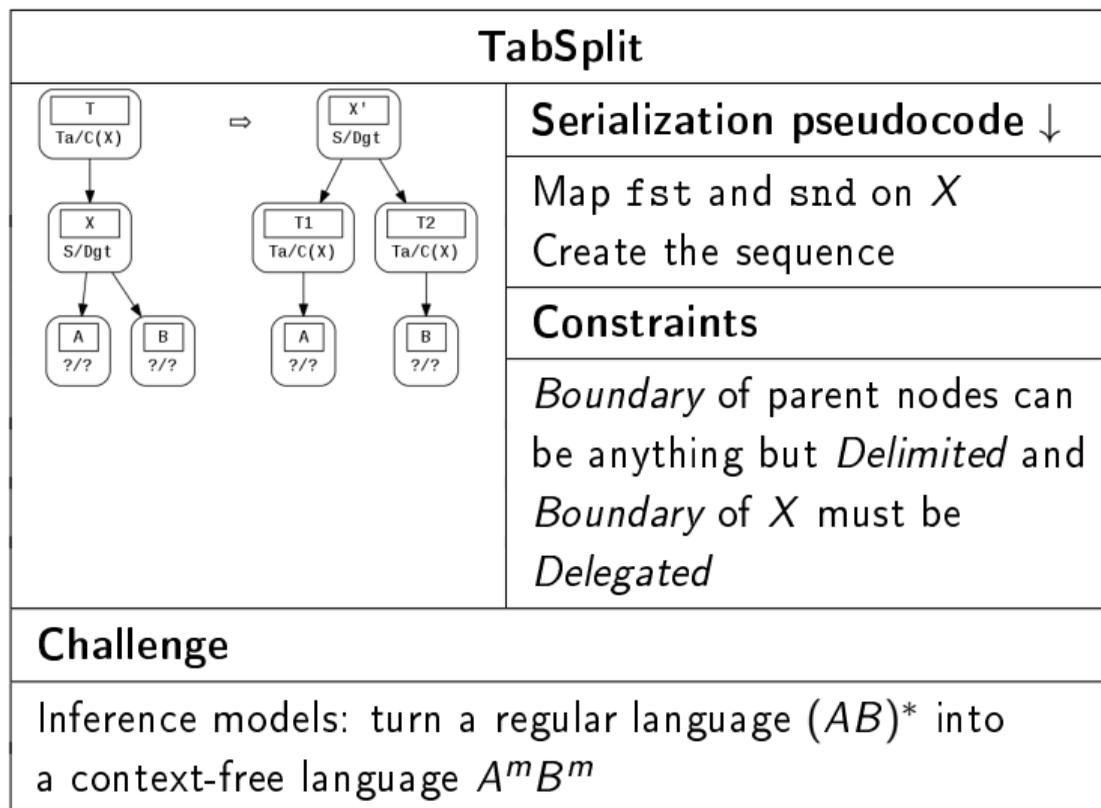
Transformations

BoundaryChange	
<p>The diagram illustrates a transformation from a single node to a tree structure. On the left, a rounded rectangle contains 'X' above '?/De'. An arrow points to a central rounded rectangle containing 'X'' above 'S/Dgt'. From this central node, two arrows point down to two separate rounded rectangles. The left one contains 'Lx' above 'Te/F(4)'. The right one contains 'X' above '?/L(Lx)'.</p>	Serialization pseudocode ↑
	Measure the serialization of X Prefix the result with this length
	Constraints
	<i>Boundary</i> of parent nodes must be either <i>Delegated</i> or <i>End</i>
Challenge	
Fields delimitation: delimitation with a length field	

Transformations

ReadFromEnd	
	Serialization pseudocode ↑
	Mirror the serialization of X
	Constraints
	<i>Boundary</i> of parent nodes can be anything but <i>Delimited</i>
Challenge	
Inference models and classification: subpart of message read in reverse order	

Transformations



Transformations

ChildMove	
	Serialization pseudocode ↓
	Switch children in X
	Constraints
	<i>Boundary</i> of parent nodes can be anything but <i>Delimited</i> and no nodes inside B must depend on a node inside A
Challenge	
Classification: meaningful fields are no more at the beginning	

Résultats et limites

Expérimentations

- ▶ Mise au défis de plusieurs experts en rétro-conception de protocoles
- ▶ Confirmation de la difficulté accrue en présence d'un protocole obfusqué avec cette stratégie
- ▶ Mais aspect difficile à évaluer
- ▶ Évaluation des performances : fonction *linéaire* du nombre de transformations réalisées
- ▶ Certaines transformations sont *impossibles*

Présentation des résultats à la conférence DSN 2018 :
Specification-based Protocol Obfuscation

Travaux a venir

- ▶ Extension du nombre de transformations
- ▶ Mise en place d'un système de diversification basé sur cette approche

References I

-  N. Chomsky, *Three models for the description of language*, Information Theory, IRE Transactions on **2** (1956), no. 3, 113–124.
-  Liu Guang-Hong, Wu Gang, Zheng Tao, Shuai Jian-Mei, and Tang Zhuo-Chun, *Vulnerability analysis for x86 executables using genetic algorithm and fuzzing*, Convergence and Hybrid Information Technology, 2008. ICCIT '08. Third International Conference on, vol. 2, November 2008, pp. 491–497.
-  John E. Hopcroft and Jeffrey D. Ullman, *Formal languages and their relation to automata*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1969.
-  Konstantinos Kemalis and Theodoros Tzouramanis, *Sql-ids: A specification-based approach for sql-injection detection*, Proceedings of the 2008 ACM Symposium on Applied Computing (New York, NY, USA), SAC '08, ACM, 2008, pp. 2153–2158.
-  P.S. Landweber, *Decision problems of phrase-structure grammars*, Electronic Computers, IEEE Transactions on **EC-13** (1964), no. 4, 354–362.

References II

-  Michel Latteux and Paavo Turakainen, *On characterizations of recursively enumerable languages*, Acta Informatica **28** (1990), no. 2, 179–186 (English).
-  Niels Heinen Michal Zalewski and Sebastian Roschke, *skipfish: web application security scanner*.
-  Jesse Rudermann, *Introducing jsfunfuzz*, 2007.

Injections sur une phrase d'un langage algébrique

Grammaire

$S \rightarrow a N \bar{a}$	$S_1 \rightarrow N \bar{a}$
$S \rightarrow b M \bar{b}$	$S_1 \rightarrow R \bar{a} o \bar{o}$
$S \rightarrow a R \bar{a} o \bar{o}$	$S_{1,2} \rightarrow N_2 \bar{a}$
$N \rightarrow b x \bar{b}$	$S_{1,2} \rightarrow R_2 \bar{a} o \bar{o}$
$N \rightarrow b M \bar{b}$	$R_2 \rightarrow x \bar{b}$
$N \rightarrow c P \bar{c}$	$N_2 \rightarrow x \bar{b}$
$R \rightarrow b x \bar{b}$	$N_2 \rightarrow M \bar{b}$
$P \rightarrow d x \bar{d}$	$S_{1,2,3} \rightarrow N_2$
$M \rightarrow d x \bar{d}$	$S_{1,2,3,4} \rightarrow N_{2,4}$
$M \rightarrow d \bar{d} v e \bar{e}$	$N_{2,4} \rightarrow x$
	$N_{2,4} \rightarrow M$

Phrase avec un point d'injection

$a b x \bar{b} \bar{a}$

Langage

$a c d x \bar{d} \bar{c} \bar{a}$
 $a b x \bar{b} \bar{a} o \bar{o}$
 $a b d \bar{d} v e \bar{e} \bar{b} \bar{a}$
 $a b d x \bar{d} \bar{b} \bar{a}$
 $a b x \bar{b} \bar{a}$
 $b d \bar{d} v e \bar{e} \bar{b}$
 $b d x \bar{d} \bar{b}$

Injections sur une phrase d'un langage algébrique

Grammaire

$S \rightarrow a N \bar{a}$	$S_1 \rightarrow N \bar{a}$
$S \rightarrow b M \bar{b}$	$S_1 \rightarrow R \bar{a} o \bar{o}$
$S \rightarrow a R \bar{a} o \bar{o}$	$S_{1,2} \rightarrow N_2 \bar{a}$
$N \rightarrow b x \bar{b}$	$S_{1,2} \rightarrow R_2 \bar{a} o \bar{o}$
$N \rightarrow b M \bar{b}$	$R_2 \rightarrow x \bar{b}$
$N \rightarrow c P \bar{c}$	$N_2 \rightarrow x \bar{b}$
$R \rightarrow b x \bar{b}$	$N_2 \rightarrow M \bar{b}$
$P \rightarrow d x \bar{d}$	$S_{1,2,3} \rightarrow N_2$
$M \rightarrow d x \bar{d}$	$S_{1,2,3,4} \rightarrow N_{2,4}$
$M \rightarrow d \bar{d} v e \bar{e}$	$N_{2,4} \rightarrow x$
	$N_{2,4} \rightarrow M$

Phrase avec un point d'injection

a b x \bar{b} \bar{a}

Langage

a c d x \bar{d} \bar{c} \bar{a}
a b x \bar{b} \bar{a} o \bar{o}
a b d \bar{d} v e \bar{e} \bar{b} \bar{a}
a b d x \bar{d} \bar{b} \bar{a}
a b x \bar{b} \bar{a}
b d \bar{d} v e \bar{e} \bar{b}
b d x \bar{d} \bar{b}

Injections sur une phrase d'un langage algébrique

Grammaire

$S \rightarrow a N \bar{a}$	$S_1 \rightarrow N \bar{a}$
$S \rightarrow b M \bar{b}$	$S_1 \rightarrow R \bar{a} o \bar{o}$
$S \rightarrow a R \bar{a} o \bar{o}$	$S_{1,2} \rightarrow N_2 \bar{a}$
$N \rightarrow b x \bar{b}$	$S_{1,2} \rightarrow R_2 \bar{a} o \bar{o}$
$N \rightarrow b M \bar{b}$	$R_2 \rightarrow x \bar{b}$
$N \rightarrow c P \bar{c}$	$N_2 \rightarrow x \bar{b}$
$R \rightarrow b x \bar{b}$	$N_2 \rightarrow M \bar{b}$
$P \rightarrow d x \bar{d}$	$S_{1,2,3} \rightarrow N_2$
$M \rightarrow d x \bar{d}$	$S_{1,2,3,4} \rightarrow N_{2,4}$
$M \rightarrow d \bar{d} v e \bar{e}$	$N_{2,4} \rightarrow x$
	$N_{2,4} \rightarrow M$

Phrase avec un point d'injection

a b x \bar{b} \bar{a}

Langage

a c d x \bar{d} \bar{c} \bar{a}
a b x \bar{b} \bar{a} o \bar{o}
a b d \bar{d} v e \bar{e} \bar{b} \bar{a}
a b d x \bar{d} \bar{b} \bar{a}
a b x \bar{b} \bar{a}
b d \bar{d} v e \bar{e} \bar{b}
b d x \bar{d} \bar{b}

Injections sur une phrase d'un langage algébrique

Grammaire

$S \rightarrow a N \bar{a}$	$S_1 \rightarrow N \bar{a}$
$S \rightarrow b M \bar{b}$	$S_1 \rightarrow R \bar{a} o \bar{o}$
$S \rightarrow a R \bar{a} o \bar{o}$	$S_{1,2} \rightarrow N_2 \bar{a}$
$N \rightarrow b x \bar{b}$	$S_{1,2} \rightarrow R_2 \bar{a} o \bar{o}$
$N \rightarrow b M \bar{b}$	$R_2 \rightarrow x \bar{b}$
$N \rightarrow c P \bar{c}$	$N_2 \rightarrow x \bar{b}$
$R \rightarrow b x \bar{b}$	$N_2 \rightarrow M \bar{b}$
$P \rightarrow d x \bar{d}$	$S_{1,2,3} \rightarrow N_2$
$M \rightarrow d x \bar{d}$	$S_{1,2,3,4} \rightarrow N_{2,4}$
$M \rightarrow d \bar{d} v e \bar{e}$	$N_{2,4} \rightarrow x$
	$N_{2,4} \rightarrow M$

Phrase avec un point d'injection

$a b x \bar{b} \bar{a}$

Langage

$a c d x \bar{d} \bar{c} \bar{a}$
 $a b x \bar{b} \bar{a} o \bar{o}$
 $a b d \bar{d} v e \bar{e} \bar{b} \bar{a}$
 $a b d x \bar{d} \bar{b} \bar{a}$
 $a b x \bar{b} \bar{a}$
 $b d \bar{d} v e \bar{e} \bar{b}$
 $b d x \bar{d} \bar{b}$

Injections sur une phrase d'un langage algébrique

Grammaire

$S \rightarrow a N \bar{a}$	$S_1 \rightarrow N \bar{a}$
$S \rightarrow b M \bar{b}$	$S_1 \rightarrow R \bar{a} o \bar{o}$
$S \rightarrow a R \bar{a} o \bar{o}$	$S_{1,2} \rightarrow N_2 \bar{a}$
$N \rightarrow b x \bar{b}$	$S_{1,2} \rightarrow R_2 \bar{a} o \bar{o}$
$N \rightarrow b M \bar{b}$	$R_2 \rightarrow x \bar{b}$
$N \rightarrow c P \bar{c}$	$N_2 \rightarrow x \bar{b}$
$R \rightarrow b x \bar{b}$	$N_2 \rightarrow M \bar{b}$
$P \rightarrow d x \bar{d}$	$S_{1,2,3} \rightarrow N_2$
$M \rightarrow d x \bar{d}$	$S_{1,2,3,4} \rightarrow N_{2,4}$
$M \rightarrow d \bar{d} v e \bar{e}$	$N_{2,4} \rightarrow x$
	$N_{2,4} \rightarrow M$

Phrase avec un point d'injection

$a b x \bar{b} \bar{a}$

Langage

$a c d x \bar{d} \bar{c} \bar{a}$
 $a b x \bar{b} \bar{a} o \bar{o}$
 $a b d \bar{d} v e \bar{e} \bar{b} \bar{a}$
 $a b d x \bar{d} \bar{b} \bar{a}$
 $a b x \bar{b} \bar{a}$
 $b d \bar{d} v e \bar{e} \bar{b}$
 $b d x \bar{d} \bar{b}$

Injections sur une phrase d'un langage algébrique

Grammaire

$S \rightarrow a N \bar{a}$	$S_1 \rightarrow N \bar{a}$
$S \rightarrow b M \bar{b}$	$S_1 \rightarrow R \bar{a} o \bar{o}$
$S \rightarrow a R \bar{a} o \bar{o}$	$S_{1,2} \rightarrow N_2 \bar{a}$
$N \rightarrow b x \bar{b}$	$S_{1,2} \rightarrow R_2 \bar{a} o \bar{o}$
$N \rightarrow b M \bar{b}$	$R_2 \rightarrow x \bar{b}$
$N \rightarrow c P \bar{c}$	$N_2 \rightarrow x \bar{b}$
$R \rightarrow b x \bar{b}$	$N_2 \rightarrow M \bar{b}$
$P \rightarrow d x \bar{d}$	$S_{1,2,3} \rightarrow N_2$
$M \rightarrow d x \bar{d}$	$S_{1,2,3,4} \rightarrow N_{2,4}$
$M \rightarrow d \bar{d} v e \bar{e}$	$N_{2,4} \rightarrow x$
	$N_{2,4} \rightarrow M$

Phrase avec un point d'injection

a **b** x \bar{b} \bar{a}

Langage

a c d x \bar{d} \bar{c} \bar{a}
a b x \bar{b} \bar{a} o \bar{o}
a b d \bar{d} v e \bar{e} \bar{b} \bar{a}
a b d x \bar{d} \bar{b} \bar{a}
a b x \bar{b} \bar{a}
b d \bar{d} v e \bar{e} \bar{b}
b d x \bar{d} \bar{b}

Injections sur une phrase d'un langage algébrique

Grammaire

$S \rightarrow a N \bar{a}$	$S_1 \rightarrow N \bar{a}$
$S \rightarrow b M \bar{b}$	$S_1 \rightarrow R \bar{a} o \bar{o}$
$S \rightarrow a R \bar{a} o \bar{o}$	$S_{1,2} \rightarrow N_2 \bar{a}$
$N \rightarrow b x \bar{b}$	$S_{1,2} \rightarrow R_2 \bar{a} o \bar{o}$
$N \rightarrow b M \bar{b}$	$R_2 \rightarrow x \bar{b}$
$N \rightarrow c P \bar{c}$	$N_2 \rightarrow x \bar{b}$
$R \rightarrow b x \bar{b}$	$N_2 \rightarrow M \bar{b}$
$P \rightarrow d x \bar{d}$	$S_{1,2,3} \rightarrow N_2$
$M \rightarrow d x \bar{d}$	$S_{1,2,3,4} \rightarrow N_{2,4}$
$M \rightarrow d \bar{d} v e \bar{e}$	$N_{2,4} \rightarrow x$
	$N_{2,4} \rightarrow M$

Phrase avec un point d'injection

a **b** x \bar{b} \bar{a}

Langage

a **c** **d** x \bar{d} \bar{c} \bar{a}
a **b** x \bar{b} \bar{a} o \bar{o}
a **b** d \bar{d} v e \bar{e} \bar{b} \bar{a}
a **b** d x \bar{d} \bar{b} \bar{a}
a **b** x \bar{b} \bar{a}
b d \bar{d} v e \bar{e} \bar{b}
b d x \bar{d} \bar{b}

Injections sur une phrase d'un langage algébrique

Grammaire

$S \rightarrow a N \bar{a}$	$S_1 \rightarrow N \bar{a}$
$S \rightarrow b M \bar{b}$	$S_1 \rightarrow R \bar{a} o \bar{o}$
$S \rightarrow a R \bar{a} o \bar{o}$	$S_{1,2} \rightarrow N_2 \bar{a}$
$N \rightarrow b x \bar{b}$	$S_{1,2} \rightarrow R_2 \bar{a} o \bar{o}$
$N \rightarrow b M \bar{b}$	$R_2 \rightarrow x \bar{b}$
$N \rightarrow c P \bar{c}$	$N_2 \rightarrow x \bar{b}$
$R \rightarrow b x \bar{b}$	$N_2 \rightarrow M \bar{b}$
$P \rightarrow d x \bar{d}$	$S_{1,2,3} \rightarrow N_2$
$M \rightarrow d x \bar{d}$	$S_{1,2,3,4} \rightarrow N_{2,4}$
$M \rightarrow d \bar{d} v e \bar{e}$	$N_{2,4} \rightarrow x$
	$N_{2,4} \rightarrow M$

Phrase avec un point d'injection

a b x \bar{b} \bar{a}

Langage

a c d x \bar{d} \bar{c} \bar{a}
a b x \bar{b} \bar{a} o \bar{o}
a b d \bar{d} v e \bar{e} \bar{b} \bar{a}
a b d x \bar{d} \bar{b} \bar{a}
a b x \bar{b} \bar{a}
b d \bar{d} v e \bar{e} \bar{b}
b d x \bar{d} \bar{b}

Injections sur une phrase d'un langage algébrique

Grammaire

$S \rightarrow a N \bar{a}$	$S_1 \rightarrow N \bar{a}$
$S \rightarrow b M \bar{b}$	$S_1 \rightarrow R \bar{a} o \bar{o}$
$S \rightarrow a R \bar{a} o \bar{o}$	$S_{1,2} \rightarrow N_2 \bar{a}$
$N \rightarrow b x \bar{b}$	$S_{1,2} \rightarrow R_2 \bar{a} o \bar{o}$
$N \rightarrow b M \bar{b}$	$R_2 \rightarrow x \bar{b}$
$N \rightarrow c P \bar{c}$	$N_2 \rightarrow x \bar{b}$
$R \rightarrow b x \bar{b}$	$N_2 \rightarrow M \bar{b}$
$P \rightarrow d x \bar{d}$	$S_{1,2,3} \rightarrow N_2$
$M \rightarrow d x \bar{d}$	$S_{1,2,3,4} \rightarrow N_{2,4}$
$M \rightarrow d \bar{d} v e \bar{e}$	$N_{2,4} \rightarrow x$
	$N_{2,4} \rightarrow M$

Phrase avec un point d'injection

$a b x \bar{b} \bar{a}$

Langage

$a c d x \bar{d} \bar{c} \bar{a}$
 $a b x \bar{b} \bar{a} o \bar{o}$
 $a b d \bar{d} v e \bar{e} \bar{b} \bar{a}$
 $a b d x \bar{d} \bar{b} \bar{a}$
 $a b x \bar{b} \bar{a}$
 $b d \bar{d} v e \bar{e} \bar{b}$
 $b d x \bar{d} \bar{b}$