Sécurité de *Remote Desktop Protocol*

Aurélien Bordes, Arnaud Ebalard,
Raphaël Rigo
prenom.nom@ssi.gouv.fr

SSTIC 2012

Agence Nationale de la Sécurité
des Systèmes d'Information
51, boulevard de la Tour-Maubourg
75700 Paris 07 SP

Contexte

Solutions d'administration à distance sous Windows :

- VNC : historique, peu sécurisé
- RDP
- RPC : via la *MMC*, limité aux produits MS
- solutions propriétaires

Critères de choix :

- l'environnement de déploiement
- les fonctionnalités
- la sécurité (accessoirement)

Limites de la présentation

Évolutions :

- RDP est initialement un protocole de déport d'affichage ...
- ...étendu avec de nouvelles fonctionnalités au fil des versions ...
- ...maintenant au cœur d'une architecture de services avec 2008 R2.

Couverture de l'étude :

- La sécurité du cœur du protocole RDP

Hors étude :

- La sécurité des architectures RDS (*Remote Desktop Services*)
- La sécurité intrinsèque des fonctionnalités et extensions

Windows Server 2008 R2: Remote Desktop Services Component Architecture

Acronyms

AD DS
Active Directory Domain Services
APP-V
Application Virtualization
CAL
client access license
GPU
graphics processing unit
MSI
Microsoft Installer Package
RDC
Remote Desktop Connection
RDS
Remote Desktop Services
RDP
Remote Desktop Protocol
RD Connection Broker
Remote Desktop Connection Broker
RD Gateway
Remote Desktop Gateway
RD Session Host
Remote Desktop Session Host
RD Virtualization Host
Remote Desktop Virtualization Host
RD Web Access
Remote Desktop Web Access
ROI
Return On Investment
SP1
Service Pack 1
SCVMM
System Center Virtual Machine Manager
SSE2
Streaming SIMD Extensions 2
SLAT
Second-Level Address Translation
SSL
Secure Sockets Layer
VDI
Virtual Desktop Infrastructure
V-GPU
virtual graphics processing unit
VMM
Virtual Machine Manager

Windows Server 2008 R2
Remote Desktop Services

Remote Desktop Session Host

1

An RD Session Host server hosts Windows-based programs or the full Windows desktop for Remote Desktop Services clients. Users can connect to an RD Session Host server to run programs, to save files, and to use network resources on that server. Users can access an RD Session Host server by using RD Connection Broker, RD Web Access, or RemoteApp and Desktop Connection.

RD Session Host server farms
You can set up a load-balanced RD Session Host server farm to scale the performance of a single RD Session Host server by distributing Remote Desktop Services sessions across multiple servers.

RD Session Host
RemoteApp programs
When a user accesses a RemoteApp program on an RD Session Host server, the program runs on the server. Each user sees only their individual session. The session is managed transparently by Remote Desktop Services and is independent of any other client session.

App-V for RDS
App-V for RDS provides a method to stream and isolate applications to RD Session Host servers. They can be published directly to the client computer. This also reduces the need to set up multiple isolated server farms.

Session-based desktops
App-V applications
Remote Desktop Session Host

RD Web Access
RD Connection Broker
Additional Remote Desktop Services infrastructure components
Active Directory Domain Services
RD Gateway

Deliver

Session-based desktops delivery
An RD Session Host server can deliver a user desktop session to any designated user in the network.

RD Session Host configuration
① Configure Network Level Authentication for the RD Session Host server.
② Configure the Remote Desktop Users group to give users and groups permissions to remotely connect.
③ Configure an RD Session Host server running Windows Server 2008 R2 to use at least one Remote Desktop license server.
Remote Desktop Connection client configuration
The RD Session Host server can be configured so that users connecting to a remote session can use some of the following functionality:
① Audio recording redirection
② Windows Aero experience
③ Remote computer's audio and video playback redirection

RemoteApp programs delivery
RemoteApp enables administrators to make programs that are accessed remotely through an RD Session Host server appear as if they are running on the client computer. Instead of being presented to the user in the desktop of the RD Session Host server, the RemoteApp program is integrated with the client computer.
① Install the RD Session Host role service on the designated server before deploying applications that will need to run as RemoteApp programs.
② Install related applications, or applications that have dependencies on other local applications, on the same RD Session Host server.
③ Deploy applications on separate RD Session Host servers in the following scenarios:
- Applications have compatibility issues that affect other programs.
- Resource intensive applications that could affect performance when running multiple instances on the same server.

Accessing RemoteApp programs
Users access RemoteApp programs in the following ways:
① Remote Desktop Protocol (.rdp) file
② RD Web Access Web portal
③ RemoteApp And Desktop Connection by using the Windows 7 Start menu

Client requirements for accessing RemoteApp programs
① To access RemoteApp programs, the client computer must be running at least RDC 6.1.
② To access RemoteApp programs, the client computer must be running either Windows 7, Windows Vista with SP1, or Windows XP with SP3.
③ To access RemoteApp and Desktop Connection through the Start menu requires Windows 7.

Remote Desktop Connection client

Remote Desktop Services Architecture

Remote Desktop Services provides a virtualization platform for accelerating and extending desktop and application deployments from the data center to any device. It provides an extensible platform for a Virtual Desktop Infrastructure.

Enumerating programs and desktops for users
1. Client queries and enumerates RemoteApp programs, session-based desktops, and virtual desktops.
2. RD Web Access queries for RemoteApp programs, session-based desktops, and virtual desktops from the RD Connection Broker server.
3. If using a personal virtual desktop, the assignment is verified by using AD DS.
4. RemoteApp programs and session-based desktops are aggregated from multiple RD Session Host servers.
5. For Windows 7 computers, icons are presented on the Start menu for RemoteApp programs, session-based desktops, and virtual desktops. For computers not running Windows 7, the icons are available by using RD Web Access.

Connecting to RemoteApp programs or session-based desktops
A Client connects to RemoteApp programs or session-based desktops.

Connecting a client computer to a virtual desktop
A User requests a connection to a virtual desktop.
B The RDP over HTTPS connection request is routed through the RD Gateway server to the RD Session Host server running in redirection mode.
C The RD Session Host server running in redirection mode informs the RD Connection Broker server that a virtual desktop is needed and waits for the IP address of the virtual desktop.
D The RD Connection Broker server orchestrates the virtual desktop and returns connection information to the RD Session Host server running in redirection mode.
E The RD Session Host server running in redirection mode sends the client computer a redirection packet containing the IP address of the virtual desktop and any connection information.
F The client computer connects to the virtual desktop.

Full-Fidelity user experience
Bi-directional audio · Windows Media Player redirection · Multiple monitor support · RemoteFX

Enhanced functionality

RD Session Host
Hosts centralized RemoteApp programs and session-based desktops.

RD Virtualization Host
Hosts centralized virtual desktops using servers running Hyper-V.

Personal Virtual Desktop
Dedicated, exclusive virtual desktop per user

Virtual Desktop Pool
Shared, identical virtual desktops for users

RemoteApp programs
Session-based desktops
RD Session Host
AD DS
RD Virtualization Host
RD Session Host server running in redirection mode
RD Connection Broker
We recommend installing the RD Connection Broker role service and the RD Session Host server running in redirection mode on the same server.
A Remote Desktop license server is required to implement Remote Desktop Services.
RD Web Access
RD Gateway
Tools to assist with application and virtual machine management
Application Virtualization for Remote Desktop Services 4.6
RD Session Host or RD Virtualization Host
App-V simplifies management of virtual desktops and applications. By using App-V with Remote Desktop Services, you can:
① Increase server farm ROI by consolidating servers.
② Minimize application conflicts and regression testing.
③ Accelerate application deployment.
④ Reduce deployment risk.
⑤ Simplify profile management.
⑥ Simplify base images with dynamic application provisioning to clients.
⑦ Minimize application duplication.
Corporate network
Perimeter network
SSL/TLS
SSL/TLS
Internet
Remote Desktop Connection client

Remote Desktop Virtualization Host

3

Understanding Remote Desktop Virtualization Host
Remote Desktop Virtualization Host is a new Remote Desktop Services role service included with Windows Server 2008 R2. RD Virtualization Host integrates with Hyper-V to provide virtual machines that can be used as personal virtual desktops or virtual desktop pools. User accounts can be assigned a unique personal virtual desktop or be redirected to a virtual desktop pool where a virtual desktop is dynamically assigned. RD Virtualization Host is an essential component to the VDI solution offered by Microsoft.

Remote Desktop Connection client
Users requesting connection or reconnection to virtual desktops
RD Web Access
Website with RemoteApp programs and virtual desktops
RD Gateway
User requests remote access from the Internet
When the RD Virtualization Host role service is installed, Server Manager checks to see if Hyper-V is installed. If Hyper-V is not installed, Server Manager will install it.

Required RDS infrastructure
RD Session Host server running in redirection mode
An RD Session Host server running in redirection mode redirects an RDP client connection to a virtual machine.

RD Connection Broker
An RD Connection Broker server is used for coordinating the availability of virtual desktops for users and ensuring that a virtual machine is running before a user connection is established.
① When using more than one RD Virtualization Host server, the RD Connection Broker server also determines which RD Virtualization Host server to use.
② The RDS Connector for VMM enables Remote Desktop Services in Windows Server 2008 R2 to use the virtual machine dynamic placement capabilities of SCVMM 2008 R2 for personal virtual desktops by enabling communication between an RD Connection Broker server and a VMM server.

An RD Virtualization Host server has the following functions:
① Monitoring virtual desktop sessions and reporting these sessions to the RD Connection Broker server.
② Preparing the virtual machine for a Remote Desktop connection when requested by the RD Connection Broker server.

RD Virtualization Host configuration
① To deploy personal virtual desktops, the schema for the Active Directory forest must be at least Windows Server 2008.
② Ensure that the RD Virtualization Host server meets the Hyper-V installation prerequisites.
③ You cannot install Windows Server 2008 R2 on a virtual machine and add it to a personal virtual desktop or virtual desktop pool.

Virtual desktops
RD Connection Broker
RD Virtualization Host

Personal virtual desktops and virtual desktop pools
Microsoft VDI stores and runs desktop workloads including a Windows client operating system, applications, and data in a virtual machine running on a Hyper-V server. In Windows Server 2008 R2 we support two VDI deployment scenarios: personal virtual desktops and virtual desktop pools. These two scenarios present two different models of assigning virtual machines to end users.

When delivering your own VDI architecture, it is important to understand the following considerations:
① The same Hyper-V server can be used to deploy personal virtual desktops and virtual desktop pools.
② Administrators can minimize the servicing of virtual desktops for both personal virtual desktops and virtual desktop pools by separating the user state from the image. You can separate the user state from the image by using roaming user profiles and folder redirection.
③ Applications may be managed outside the image by using App-V.

When delivering your own VDI architecture, it is important to understand the following requirements:
① A virtual machine cannot be both a member of a virtual desktop pool and assigned to a user as a personal virtual desktop.
② Virtual desktops can only use Windows client operating systems.

RD Virtualization Host
Personal virtual desktops
RD Virtualization Host
Virtual desktop pools

Personal virtual desktops
Personal virtual desktops are virtual machine-based, centralized desktops for individual users and are provisioned from the RD Virtualization Host server.
① All images share the same operating system and configuration.
② Dedicated virtual desktops allow administrative privileges.
③ A virtual desktop is persistent and customizable.

RD Virtualization Host
Provision a personal virtual desktop
Assign the virtual desktop to a domain user account
Users can access the personal virtual desktop by using RemoteApp and Desktop Connection or RD Web Access.
Remote Desktop Connection client
AD DS
Redirect user data
All user data and profile information can be stored on the personal virtual desktop, or the profiles and folders can be redirected.
File server

Virtual desktop pools
Virtual desktop pools provide virtual machine-based, centralized desktops based on a pool of virtual machines that are shared by multiple users.
① All images share the same operating system and configuration.
② A virtual machine can be a member of only one virtual desktop pool at a time.
③ If rollback is enabled, changes to the virtual desktop are discarded when the user logs off. User state is not discarded if the user state and data is stored outside of the virtual desktop.

Temporarily assign a virtual desktop from the virtual desktop pool to the user
Applications can either be installed in the base image for all users or can be provisioned with App-V.
RD Virtualization Host
Remote Desktop Connection client
Redirect user data
Use roaming profiles and folder redirection to centrally manage user data and settings.
File server
A virtual desktop pool is a group of identically configured virtual machines on an RD Virtualization Host server. Users can access the virtual machines in a virtual desktop pool by using RemoteApp and Desktop Connection or RD Web Access.

Remote Desktop Licensing

2

Remote Desktop Licensing manages the Remote Desktop Services client access licenses and VDI Suite licenses that are required for each device or user to connect to session-based desktops, RemoteApp programs, or virtual desktops.
When a client connects to an RD Session Host server, the RD Session Host server determines if a license is needed. The RD Session Host server then requests an RDS CAL or VDI Suite license from a Remote Desktop license server on behalf of the client. If an appropriate license is available from a license server, the RDS CAL or VDI Suite license is issued to the client, and the client will be able to connect to the session-based desktop, RemoteApp program, or virtual desktop.

Client connects to virtual desktop
Remote Desktop Connection
Client requests license
Request license for client
Client receives license
Remote Desktop license server issues license
Client connects to session-based desktop
RD Session Host
Remote Desktop license server
Each user or device that connects to the RD Session Host server must have its own license.
Active Directory Domain Services must be installed for Per User licensing. The RD Session Host server queries AD DS for the license assigned to the user.
There is a licensing grace period (120 days) during which no license server is required.
To use Remote Desktop Services, you must have at least one license server. For small deployments, the RD Session Host role service and the RD Licensing role service can be on the same computer. For larger deployments, it is recommended that the RD Licensing role service be installed on a separate computer from the RD Session Host role service.

Microsoft Virtual Desktop Infrastructure Licensing
For VDI implementations, the following options are available:

VDI Standard Suite
The VDI Standard Suite is for deploying the basic infrastructure for VDI. Hyper-V Server 2008 R2 provides the virtualization platform; management of the virtual infrastructure is provided by system center components.

VDI Premium Suite
The VDI Premium Suite is for customers wanting additional flexibility from their VDI environment. It includes all of the VDI Standard Suite options, including unrestricted Remote Desktop Services, enabling customers to deliver both session-based desktops and virtual desktops.

Remote Desktop Connection Broker

4

Remote Desktop Connection Broker provides a single, personalized, and aggregated view of RemoteApp programs, session-based desktops, and virtual desktops to users. RD Connection Broker supports load balancing and reconnection to existing sessions on virtual desktops, session-based desktops, and RemoteApp programs.

Remote Desktop Connection client
Active Directory Domain Services
RD Web Access (optional)
RD Gateway (optional)
Required RDS infrastructure

RD Session Host server farm
Users can connect to new or established sessions in a load-balanced RD Session Host server farm.
RD Gateway is not required if users are only connecting from access from the Internet.

RD Connection Broker
RD Connection Broker connects or reconnects a client computer to either a session-based desktop, virtual desktop, or RemoteApp program.
The RD Connection Broker server stores the following session state information:
① Session IDs
② User names
③ Name of the RD Session Host server where each session resides
Connect to
Reconnect to
RemoteApp programs
Session-based desktops
NEW session-based desktops or RemoteApp programs
EXISTING session-based desktops or RemoteApp programs
Virtual desktops
NEW virtual desktops
EXISTING virtual desktops
RD Session Host
RD Virtualization Host
RD Session Host server running in redirection mode
In a VDI environment, the RD Session Host server running in redirection mode communicates with the RD Connection Broker server.
An RD Session Host server running in redirection mode is not available to publish RemoteApp programs or host new remote sessions.
① When a user requests connection to a virtual machine, the RD Session Host server running in redirection mode queries the RD Connection Broker server.
② The RD Connection Broker server provisions a virtual machine for the user and returns connection details to the RD Session Host server running in redirection mode.
③ The server then redirects the RDP client to connect to the appropriate virtual machine.

Remote Desktop Web Access

5

A Remote Desktop Web Access server provides users with a customizable web portal for accessing session-based desktops, virtual desktops, and RemoteApp programs.

Remote Desktop Connection
Client queries an RD Web Access server over HTTPS
Internet Explorer web browser
RemoteApp and Desktop Connection

RD Web Access
RD Web Access enables the following:
① Access to RemoteApp programs, session-based desktops, and virtual desktops from the RD Web Access website.
② Display resources from more than one farm, or a combination of farm and individual servers.
③ Filter the view on a per-user basis so that each user sees only the authorized programs.

RD Connection Broker
The RD Connection Broker server authenticates the user and retrieves a list of all RemoteApp programs, session-based desktops, and virtual desktops and presents it to the RD Web Access server.

RD Session Host
Assigned session-based desktops and RemoteApp programs
RD Virtualization Host
Assigned personal virtual desktops and virtual desktop pools

Client connection options
Connect by using a web browser
Connect using a web browser by signing into the RD Web Access website. In this case, a computer that is running Windows 7 is not required.
Connect by using RemoteApp and Desktop Connection
All resources will show up in the Start menu under All Programs in a folder called RemoteApp and Desktop Connections. A computer running Windows 7 is required.

Remote Desktop Connection webpage
A user can view all RemoteApp programs and virtual desktops published to that user.
RemoteApp programs
Session-based desktops
Virtual desktops
Users only enter their user name and password once when connecting to a RemoteApp program by using RD Web Access.
Group Policy can assist in the deployment of RemoteApp programs for enterprise-sized businesses.
For enterprise deployments, the RD Web Access server can be installed on a front-end web server to service multiple RD Session Host servers on the back end.
Personal virtual desktops and virtual desktop pools are hosted on the RD Virtualization Host server.
The RD Virtualization Host server can be configured so that each user in the organization is assigned to a unique virtual desktop, or are redirected to a virtual desktop pool where a virtual desktop is dynamically assigned.

Remote Desktop Gateway

6

The Remote Desktop Gateway role service in Windows Server 2008 R2 enables compatible devices to securely connect over the Internet to RD Session Host servers or RD Virtualization Host servers behind a corporate firewall. Network resources can be any authorized session-based desktops, RemoteApp programs, or virtual desktops.

RD Gateway uses the RDP over HTTPS to establish a secure, encrypted connection between remote users on the Internet and the internal network resources on which their production applications run.
Remote Desktop Connection client
RDP / HTTPS
External firewall
(Port 443 OPEN)
Perimeter network
RD Gateway
Internal firewall
(Port 3389 OPEN)
Internet
For redundancy and load balancing, RD Gateway can be configured as a cluster.
RD Session Host
Session-based desktops and RemoteApp programs
RDP
RD Virtualization Host
Virtual desktops
Client access
Client computers that have Remote Desktop enabled
Internal network

RD Gateway configuration
① Install the RD Gateway role service.
② Obtain and configure an SSL-compatible X.509 certificate on the RD Gateway server.
③ Create a Remote Desktop connection authorization policy (RD CAP).
④ Create a Remote Desktop resource authorization policy (RD RAP).
⑤ Configure the RDC client to use RD Gateway.

Users can securely access session-based desktops, RemoteApp programs, and virtual desktops from the Internet.

Microsoft RemoteFX

Microsoft RemoteFX delivers a rich user experience for VDI by using graphics processing units that are present on the server and shared across multiple virtual desktops.
RemoteFX contains the following elements:
① Host-side V-GPU rendering and capture for 3D
② Improved encode/decode within RDP
③ Generic USB redirection
④ Enablement of ultra-thin client types

RemoteFX for RD Virtualization Host
RemoteFX enables:
① Rendered rich 3D content by using a virtualized GPU with Windows 7 Enterprise with SP1 or Windows 7 Ultimate with SP1
② Generic USB redirection
③ Improved encode/decode that improves efficiency and user experience
④ Ultra-thin client support
RemoteFX support requirements:
① GPU in the RD Virtualization Host server for 3D rendering
② Optional dedicated RemoteFX encode hardware
③ SLAT-capable processor in the RemoteFX server

RemoteFX for RD Session Host
RemoteFX enables:
① Improved encode/decode that improves efficiency and user experience
② Ultra-thin client support
RemoteFX support requirements:
① Optional dedicated RemoteFX encode hardware
② SSE2-capable processor in the RD Session Host server

RemoteFX-enabled virtual desktops by using RDP
RDP
RemoteFX integrates with RDP, which enables shared encryption, authentication, management, and device support.
Remote Desktop Connection client
RemoteFX supports a range of client devices including rich clients, thin clients, and ultra-thin clients.
① The software client is part of Remote Desktop Connection that is included with Windows client and server operating systems and Windows Embedded.
② An RDP client with RemoteFX can run as software or by using a RemoteFX hardware decoder.

Author: Martin McClean with contributions from Brian Lich and Thomas Willingham (Microsoft Windows Server Information Experience Team, Redmond WA email: rdsua@microsoft.com)

Windows Server 2008 R2: Remote Desktop Services Component Architecture

Vous êtes ici !

Acronyms

AD DS
Active Directory Domain Services
APP-V
Application Virtualization
CAL
client access license
GPU
graphics processing unit
MSI
Microsoft Installer Package
RDC
Remote Desktop Connection
RDS
Remote Desktop Services
RDP
Remote Desktop Protocol
RD Connection Broker
Remote Desktop Connection Broker
RD Gateway
Remote Desktop Gateway
RD Session Host
Remote Desktop Session Host
RD Virtualization Host
Remote Desktop Virtualization Host
RD Web Access
Remote Desktop Web Access
ROI
Return On Investment
SP1
Service Pack 1
SCVMM
System Center Virtual Machine Manager
SSE2
Streaming SIMD Extensions 2
SLAT
Second-Level Address Translation
SSL
Secure Sockets Layer
VDI
Virtual Desktop Infrastructure
VGPU
virtual graphics processing unit
VMM
Virtual Machine Manager

Windows Server 2008 R2 Remote Desktop Services

Remote D…

An RD Session Host server ho… Desktop Services clients. Use… and to use network resources… Connection Broker, RD Web A…

RD Session Host server farms
You can use a load-balanced RD Session Host server farm to scale the performance of a single RD Session Host server by distributing Remote Desktop Services sessions across multiple servers.

RD Web Access | RD Connection Broker | Additional Remote Desktop Services Infrastructure components | Active Directory Domain Services | RD Gateway

Deliver

Session-based desktops delivery
An RD Session Host server can deliver a user desktop session to any designated user in the network.

RD Session Host configuration
① Configure Network Level Authentication for the RD Session Host server.
② Configure the Remote Desktop Users group to give users and groups permissions to remotely connect
③ Configure an RD Session Host server running Windows Server 2008 R2 to use at least one Remote Desktop license server
Remote Desktop Connection client configuration
The RD Session Host server can be configured so that users connecting to a remote session can use some of the following functionality:
① Audio recording redirection
② Windows Aero experience
③ Remote computer's audio and video playback redirection

RemoteApp programs delivery
RemoteApp enables administrators to make programs that are accessed remotely through an RD Session Host server appear as if they are running on the client computer. Instead of being presented to the user in the desktop of the RD Session Host server, the RemoteApp program is integrated with the client computer.
① Install the RD Session Host role service on the designated server before installing any applications that will need to run as RemoteApp programs.
② Install related applications, or applications that have dependencies on other local applications, on the same RD Session Host server.
③ Deploy applications on separate RD Session Host servers in the following scenarios:
- Applications have compatibility issues that affect other programs.
- Resource intensive applications that could affect performance when running multiple instances on the same server.

Accessing RemoteApp programs
Users access RemoteApp programs in the following ways:
① Remote Desktop Protocol (.rdp) file
② RD Web Access Web portal
③ RemoteApp And Desktop Connection by using the Windows 7 Start menu

Client requirements for accessing RemoteApp programs
① To access RemoteApp programs, the client computer must be running at least RDC 6.1.
② To access RemoteApp programs, the client computer must be running either Windows 7, Windows Vista with SP1, or Windows XP with SP3.
③ To access RemoteApp and Desktop Connection through the Start menu requires Windows 7.

Remote Desktop Connection client

Remote Desktop Services Architecture

…Desktop Services provides a virtualization platform for accelerating and extending desktop and application deployments from the data center to any device. It provides …ible platform for a Virtual Desktop Infrastructure.

…ning programs and desktops for users
…t queries and enumerates RemoteApp …rams, session-based desktops, and virtual …tops.
…Web Access queries for RemoteApp programs, …on-based desktops, and virtual desktops from …RD Connection Broker server.
…ing a personal virtual desktop, the assignment is …ed by using AD DS.
…oteApp programs and session-based …ktops are aggregated from multiple RD …ion Host servers.
…7 computers, icons are presented …the Start menu for RemoteApp programs, …session-based desktops, and virtual desktops. …or computers not running Windows 7, the …icons are available by using …b Access.

RD Session Host — Hosts centralized RemoteApp programs and session-based desktops.
RD Virtualization Host — Hosts centralized virtual desktops using servers running Hyper-V.
RemoteApp programs
Session-based desktops
Personal Virtual Desktop — Dedicated, exclusive virtual desktop per user
Virtual Desktop Pool — Shared, identical virtual desktops for users
AD DS
RD Session Host server running in redirection mode
We recommend installing the RD Connection Broker role service and the RD Session Host server running in redirection mode on the same server.
A Remote Desktop license server is required to implement Remote Desktop Services.
Tools to assist with application and virtual machine management

Connecting to RemoteApp programs or session-based …ops
A Client connects to RemoteApp programs or session-based desktops.

Connecting a client computer to a virtual desktop
A User requests a connection to a virtual desktop.
B The RDP over HTTPS connection request is routed through the RD Gateway server to the RD Session Host server running in redirection mode.
C The RD Session Host server running in redirection mode informs the RD Connection Broker server that a virtual desktop is needed and waits for the IP address of the virtual desktop.
D The RD Connection Broker server orchestrates the virtual desktop and returns connection information to the RD Session Host server running in redirection mode.
E The RD Session Host server running in redirection mode sends the client computer a redirection packet containing the IP address of the virtual desktop and any connection information.
F The client computer connects to the virtual desktop.

RD Web Access | RD Gateway | RD Connection Broker | RD Session Host | Corporate network | Perimeter network | HTTP/HTTPS | SLATP | Internet | Remote Desktop Connection client

Full-Fidelity user experience
Bi-directional audio | Windows Media Player redirection | Multiple monitor support | RemoteFX
Enhanced functionality

Application Virtualization for Remote Desktop Services 4.0
RD Session Host or RD Virtualization Host
App-V simplifies management of virtual desktops and applications. By using App-V with Remote Desktop Services, you can:
① Increase server farm ROI by consolidating servers.
② Minimize application conflicts and regression testing.
③ Accelerate application deployment.
④ Reduce deployment risk.
⑤ Simplify profile management.
⑥ Simplify base images with dynamic application provisioning to clients.
⑦ Minimize application duplication.

3 Remote Desktop Virtualization Host

Understanding Remote Desktop Virtualization Host
Remote Desktop Virtualization Host is a new Remote Desktop Services role service included with Windows Server 2008 R2. RD Virtualization Host integrates with Hyper-V to provide virtual machines that can be used as personal virtual desktops or virtual desktop pools. User accounts can be assigned a unique personal virtual desktop or be redirected to a virtual desktop pool where a virtual desktop is dynamically assigned. RD Virtualization Host is an essential component to the VDI solution offered by Microsoft.

Remote Desktop Connection client
Required RDS infrastructure
RD Session Host server running in redirection mode
An RD Session Host server running in redirection mode redirects an RDP client connection to a virtual machine.
When using more than one RD Virtualization Host server, the RD Connection Broker server also determines which RD Virtualization Host server to use.
RD Connection Broker
An RD Connection Broker server is used for coordinating the availability of virtual desktops for users and ensuring that a virtual machine is running before a user connection is established.
Users requesting connection or reconnection to virtual desktops
① The RDS Connector for VMM enables Remote Desktop Services in Windows Server 2008 R2 to use the virtual machine dynamic placement capabilities of SCVMM 2008 R2 for personal virtual desktops by enabling communication between an RD Connection Broker server and a VMM server.
RD Web Access
Website with RemoteApp programs and virtual desktops
RD Gateway
User requests remote access from the Internet
When the RD Virtualization Host role service is installed, Server Manager checks to see if Hyper-V is installed. If Hyper-V is not installed, Server Manager will install it.
An RD Virtualization Host server has the following functions:
① Monitoring virtual desktop sessions and reporting these sessions to the RD Connection Broker server.
② Preparing the virtual machine for a Remote Desktop connection when requested by the RD Connection Broker server.
RD Virtualization Host configuration
① To deploy personal virtual desktops, the schema for the Active Directory forest must be at least Windows Server 2008.
② Ensure that the RD Virtualization Host server meets the Hyper-V installation prerequisites.
③ You cannot install Windows Server 2008 R2 on a virtual machine and add it to a personal virtual desktop or virtual desktop pool.
Virtual desktops
RD Connection Broker
RD Virtualization Host

Personal virtual desktops and virtual desktop pools
Microsoft VDI stores and runs desktop workloads including a Windows client operating system, applications, and data in a virtual machine running on a Hyper-V server. In Windows Server 2008 R2 we support two VDI deployment scenarios: personal virtual desktops and virtual desktop pools. These two scenarios present two different models of assigning virtual machines to end users.

When delivering your own VDI architecture, it is important to understand the following considerations:
① The same Hyper-V server can be used to deploy personal virtual desktops and virtual desktop pools.
② Administrators can minimize the servicing of virtual desktops for both personal virtual desktops and virtual desktop pools. You can separate the user state from the image. You can separate the user state from the image by using roaming user profiles and folder redirection.
③ Applications may be managed outside the image by using App-V.

When delivering your own VDI architecture, it is important to understand the following requirements:
① A virtual machine cannot be both a member of a virtual desktop pool and assigned to a user as a personal virtual desktop.
② Virtual desktops can only use Windows client operating systems.

RD Virtualization Host — Personal virtual desktops
RD Virtualization Host — Virtual desktop pools

Personal virtual desktops
Personal virtual desktops are virtual machine-based, centralized desktops for individual users and are provisioned from the RD Virtualization Host server.
① Any virtual desktop is assigned to each user.
② Dedicated virtual desktops allow administrative privileges.
③ A virtual desktop is persistent and customizable.

RD Virtualization Host
Provision a personal virtual desktop
Assign the virtual desktop to a domain user account
Users can access the personal virtual desktop by using RemoteApp and Desktop Connection or RD Web Access.
Remote Desktop Connection client
AD DS
Redirect user data
All user data and profile information can be stored on the personal virtual desktop, or the profiles and folders can be redirected.
File server

Virtual desktop pools
Virtual desktop pools provide virtual machine-based, centralized desktops based on a pool of virtual machines that are shared by multiple users.
① All images share the same operating system and configuration.
② A virtual machine can be a member of only one virtual desktop pool at a time.
③ If rollback is enabled, changes to the virtual desktop are discarded when the user logs off. User state is not discarded if the user state and data is stored outside of the virtual desktop.

Temporarily assign a virtual desktop from the virtual desktop pool to the user
Applications can either be installed in the base image for all users or can be provisioned with App-V.
RD Virtualization Host
Remote Desktop Connection client
Redirect user data
Use roaming profiles and folder redirection to centrally manage user data and settings.
File server
A virtual desktop pool is a group of identically configured virtual machines on an RD Virtualization Host server. Users can access the virtual machines in a virtual desktop pool by using RemoteApp and Desktop Connection or RD Web Access.

2 Remote Desktop Licensing

Remote Desktop Licensing manages the Remote Desktop Services client access licenses and VDI Suite licenses that are required for each device or user to connect to session-based desktops, RemoteApp programs, or virtual desktops.
When a client connects to an RD Session Host server, the RD Session Host server determines if a license is needed. The RD Session Host server then requests an RDS CAL or VDI Suite license from a Remote Desktop license server on behalf of the client. If an appropriate license is available from a license server, the RDS CAL or VDI Suite license is issued to the client, and the client will be able to connect to the session-based desktop, RemoteApp program, or virtual desktop.

Client connects to virtual desktop
Remote Desktop Connection
Client requests license
Request license for client
Client receives license
Remote Desktop license server issues license
Client connects to session-based desktop
Each user or device that connects to the RD Session Host server must have its own license.
Active Directory Domain Services must be installed for Per User licensing. The RD Session Host server queries AD DS for the license assigned to the user.
There is a licensing grace period (120 days) during which no license server is required.
RD Session Host
Remote Desktop license server
To use Remote Desktop Services, you must have at least one license server. For small deployments, the RD Session Host role service and the RD Licensing role service can be on the same computer. For larger deployments, it is recommended that the RD Licensing role service be installed on a separate computer from the RD Session Host role service.

Microsoft Virtual Desktop Infrastructure Licensing
For VDI implementations, the following options are available:

VDI Standard Suite
The VDI Standard Suite is for deploying the basic infrastructure for VDI. Hyper-V Server 2008 R2 provides the virtualization platform; management of the virtual infrastructure is provided by system center components.

VDI Premium Suite
The VDI Premium Suite is for customers wanting additional flexibility from their VDI environment. It includes all of the VDI Standard Suite options, including unrestricted Remote Desktop Services, enabling customers to deliver both session-based desktops and virtual desktops.

4 Remote Desktop Connection Broker

Remote Desktop Connection Broker provides a single, personalized, and aggregated view of RemoteApp programs, session-based desktops, and virtual desktops to users. RD Connection Broker supports load balancing and reconnection to existing sessions on virtual desktops, session-based desktops, and RemoteApp programs.

Remote Desktop Connection client
Active Directory Domain Services
RD Web Access (optional)
RD Gateway (optional)
Required RDS infrastructure
RD Session Host server farm
Users can connect to new or established sessions in a load-balanced RD Session Host server farm.
RD Gateway is not required if users are only connecting from the Internet.
RD Connection Broker
RD Connection Broker connects or reconnects a client computer to either a session-based desktop, virtual desktop, or RemoteApp program.
The RD Connection Broker server stores the following session state information:
① Session IDs
② User names
③ Name of the RD Session Host server where each session resides
RemoteApp programs
Session-based desktops
Virtual desktops
Connect to
Reconnect to
NEW session-based desktops or RemoteApp programs
EXISTING session-based desktops or RemoteApp programs
NEW virtual desktops
EXISTING virtual desktops
RD Session Host
RD Virtualization Host
RD Connection Broker
RD Connection Broker connects or reconnects a client computer to either a personal virtual desktop or a virtual desktop pool.
RD Session Host server running in redirection mode
In a VDI environment, the RD Session Host server running in redirection mode communicates with the RD Connection Broker server.
An RD Session Host server running in redirection mode is not available to publish RemoteApp programs or host new remote sessions.
① When a user requests connection to a virtual machine, the RD Session Host server running in redirection mode queries the RD Connection Broker server.
② The RD Connection Broker server provisions a virtual machine for the user and returns connection details to the RD Session Host server running in redirection mode.
③ The server then redirects the RDP client to connect to the appropriate virtual machine.

5 Remote Desktop Web Access

A Remote Desktop Web Access server provides users with a customizable web portal for accessing session-based desktops, virtual desktops, and RemoteApp programs.

Remote Desktop Connection
Client queries an RD Web Access server over HTTPS
RemoteApp and Desktop Connection
Internet Explorer web browser

RD Web Access
RD Web Access enables the following:
① Access to RemoteApp programs, session-based desktops, and virtual desktops from the RD Web Access website.
② Display resources from more than one farm, or a combination of farm and individual servers.
③ Filter the view on a per-user basis so that each user sees only the authorized programs.

RD Connection Broker
The RD Connection Broker server authenticates the user and retrieves a list of all RemoteApp programs, session-based desktops, and virtual desktops and presents it to the RD Web Access server.

RD Session Host — Assigned session-based desktops and RemoteApp programs
RD Virtualization Host — Assigned personal virtual desktops and virtual desktop pools

Client connection options
Connect by using a web browser
Connect using a web browser by signing into the RD Web Access website. In this case, a computer that is running Windows 7 is not required.
Connect by using RemoteApp and Desktop Connection
All resources will show up in the Start menu under All Programs in a folder called RemoteApp and Desktop Connections. A computer running Windows 7 is required.

Remote Desktop Connection webpage
A user can view all RemoteApp programs and virtual desktops published to that user.
RemoteApp programs
Session-based desktops
Virtual desktops
Users only enter their user name and password once when connecting to a RemoteApp program by using RD Web Access.
Group Policy can assist in the deployment of RemoteApp programs for enterprise-sized businesses.
For enterprise deployments, the RD Web Access server can be installed on a front-end web server to service multiple RD Session Host servers on the back end.
Personal virtual desktops and virtual desktop pools are hosted on the RD Virtualization Host server.
The RD Virtualization Host server can be configured so that each user in the organization is assigned to a unique virtual desktop, or are redirected to a virtual desktop pool where a virtual desktop is dynamically assigned.

6 Remote Desktop Gateway

The Remote Desktop Gateway role service in Windows Server 2008 R2 enables compatible devices to securely connect over the Internet to RD Session Host servers or RD Virtualization Host servers behind a corporate firewall. Network resources can be any authorized session-based desktops, RemoteApp programs, or virtual desktops.

Remote Desktop Connection client
RDP / HTTPS
External firewall (Port 443 OPEN)
Perimeter network
RD Gateway
Internal firewall (Port 3389 OPEN)
Internet
For redundancy and load balancing, RD Gateway can be configured as a cluster.
RD Session Host
Session-based desktops and RemoteApp programs
RD Virtualization Host
Virtual desktops
Client access
Client computers that have Remote Desktop enabled
Internal network

RD Gateway uses the RDP over HTTPS to establish a secure, encrypted connection between remote users on the Internet and the internal network resources on which their production applications run.

RD Gateway configuration
① Install the RD Gateway role service.
② Obtain and configure an SSL-compatible X.509 certificate on the RD Gateway server.
③ Create a Remote Desktop connection authorization policy (RD CAP).
④ Create a Remote Desktop resource authorization policy (RD RAP).
⑤ Configure the RDC client to use RD Gateway.

Users can securely access session-based desktops, RemoteApp programs, and virtual desktops from the Internet.

Microsoft RemoteFX

Microsoft RemoteFX delivers a rich user experience for VDI by using graphics processing units that are present on the server and shared across multiple virtual desktops.
RemoteFX contains the following elements:
① Host-side V-GPU rendering and capture for 3D
② Improved encode/decode within RDP
③ Generic USB redirection
④ Enablement of ultra-thin client types

RemoteFX for VD Virtualization Host
RemoteFX enables:
① Rendered rich 3D content by using a virtualized GPU with Windows 7 Enterprise with SP1 or Windows 7 Ultimate with SP1
② Generic USB redirection
③ Improved encode/decode that improves efficiency and user experience
④ Ultra-thin client support
RemoteFX support requirements:
① GPU in the RD Virtualization Host server for 3D rendering
② Optional dedicated RemoteFX encode hardware
③ SLAT-capable processor in the RemoteFX server

RemoteFX for RD Session Host
RemoteFX enables:
① Improved encode/decode that improves efficiency and user experience
② Ultra-thin client support
RemoteFX support requirements:
① Optional dedicated RemoteFX encode hardware
② SSE2-capable processor in the RD Session Host server

RemoteFX-enabled virtual desktops by using RDP
RDP
RemoteFX integrates with RDP, which enables shared encryption, authentication, management, and device support.
Remote Desktop Connection client
RemoteFX supports a range of client devices including rich clients, thin clients, and ultra-thin clients.
① The software client is part of Remote Desktop Connection that is included with Windows client and server operating systems and Windows Embedded.
② An RDP client with RemoteFX can run as software or by using a RemoteFX hardware decoder.

Author: Martin McClean with contributions from Brian Lich and Thomas Willingham (Microsoft Windows Server Information Experience Team, Redmond WA email: rdsua@microsoft.com)

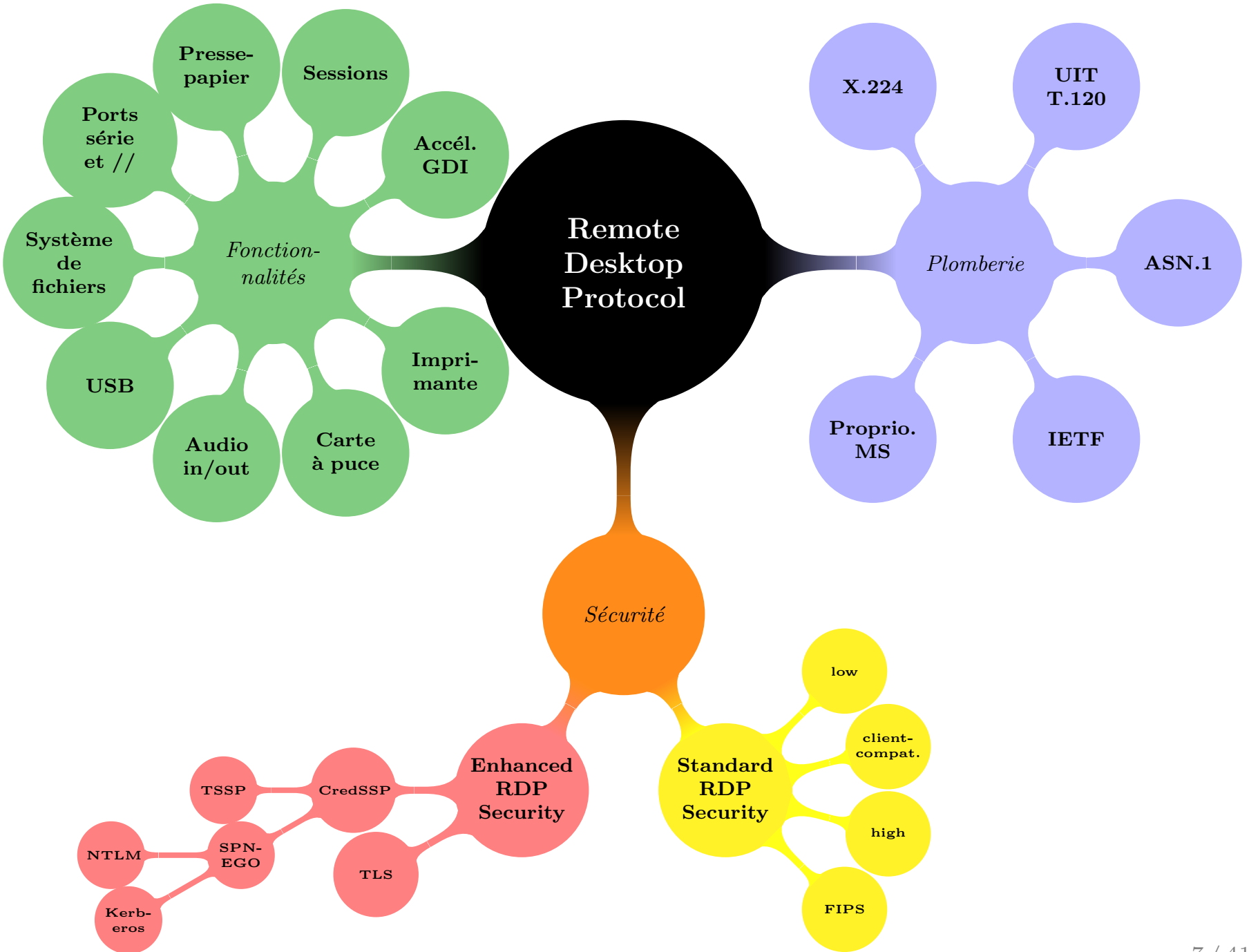
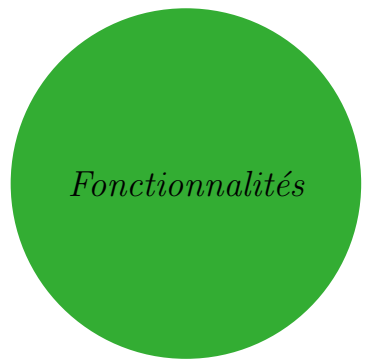
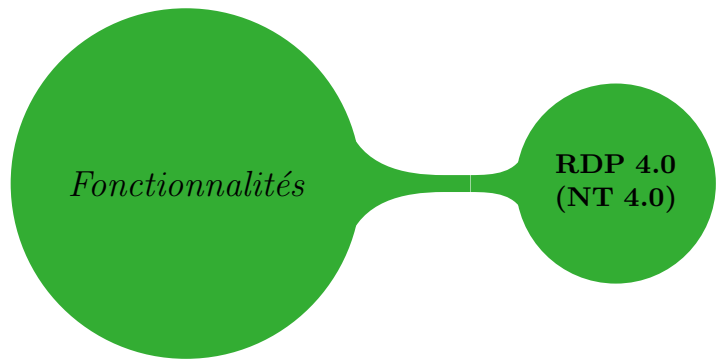
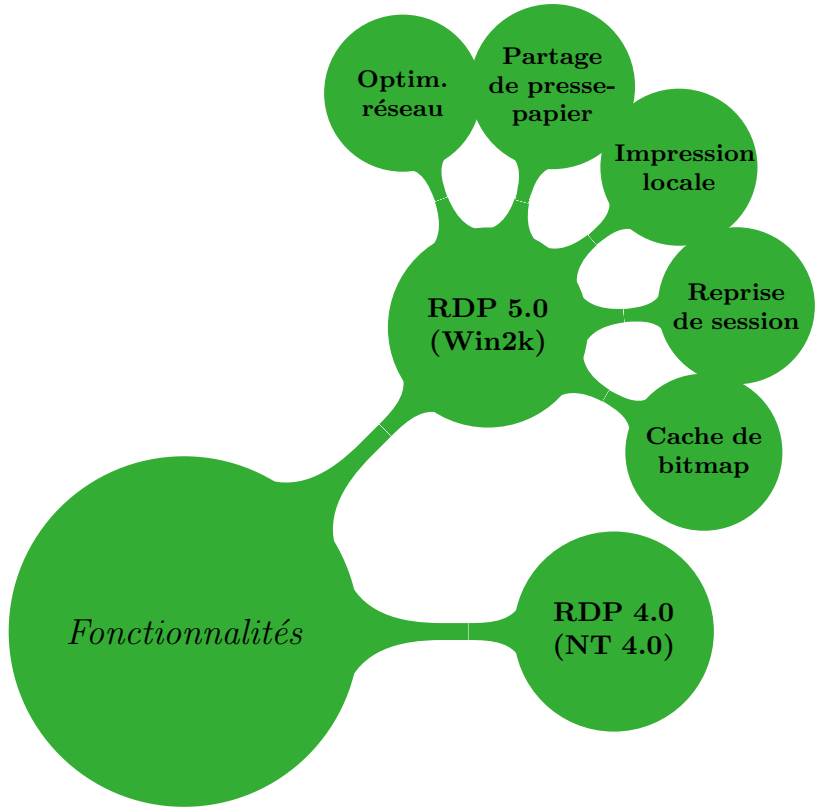
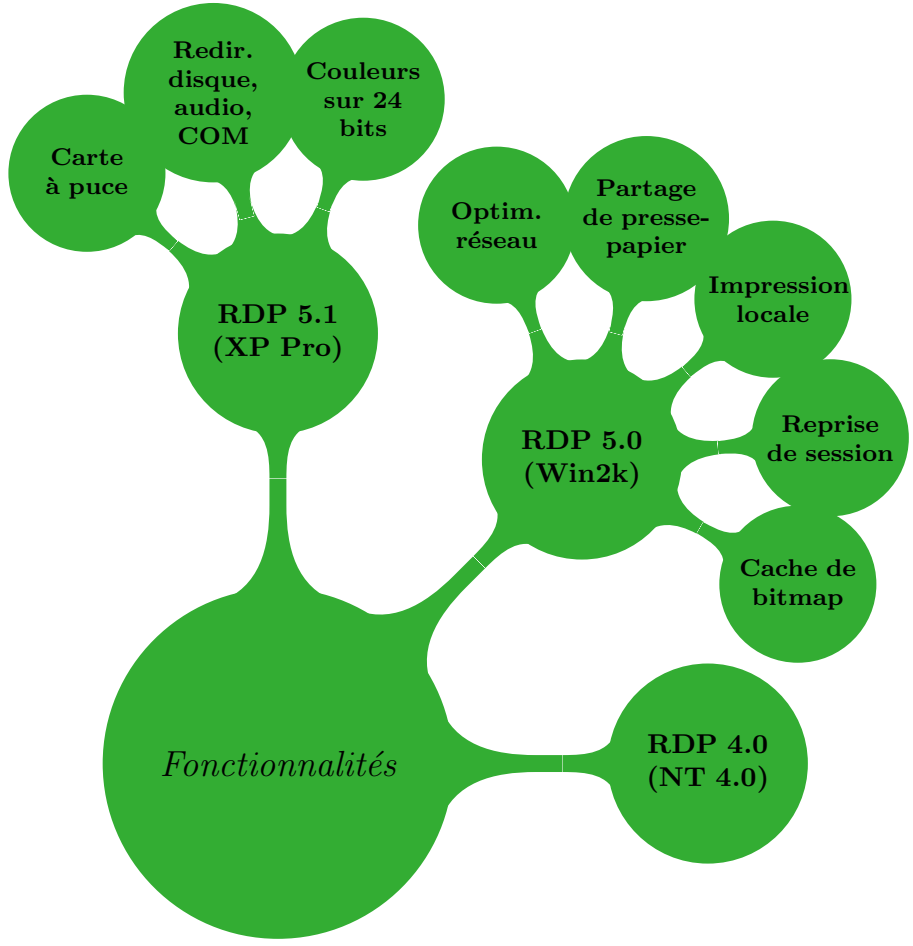

Remote Desktop Protocol
Fonction-nalités
Presse-papier
Sessions
Ports série et //
Accél. GDI
Système de fichiers
USB
Impri-mante
Audio in/out
Carte à puce
Plomberie
X.224
UIT T.120
ASN.1
Proprio. MS
IETF
Sécurité
Enhanced RDP Security
TSSP
CredSSP
NTLM
SPN-EGO
Kerb-eros
TLS
Standard RDP Security
low
client-compat.
high
FIPS

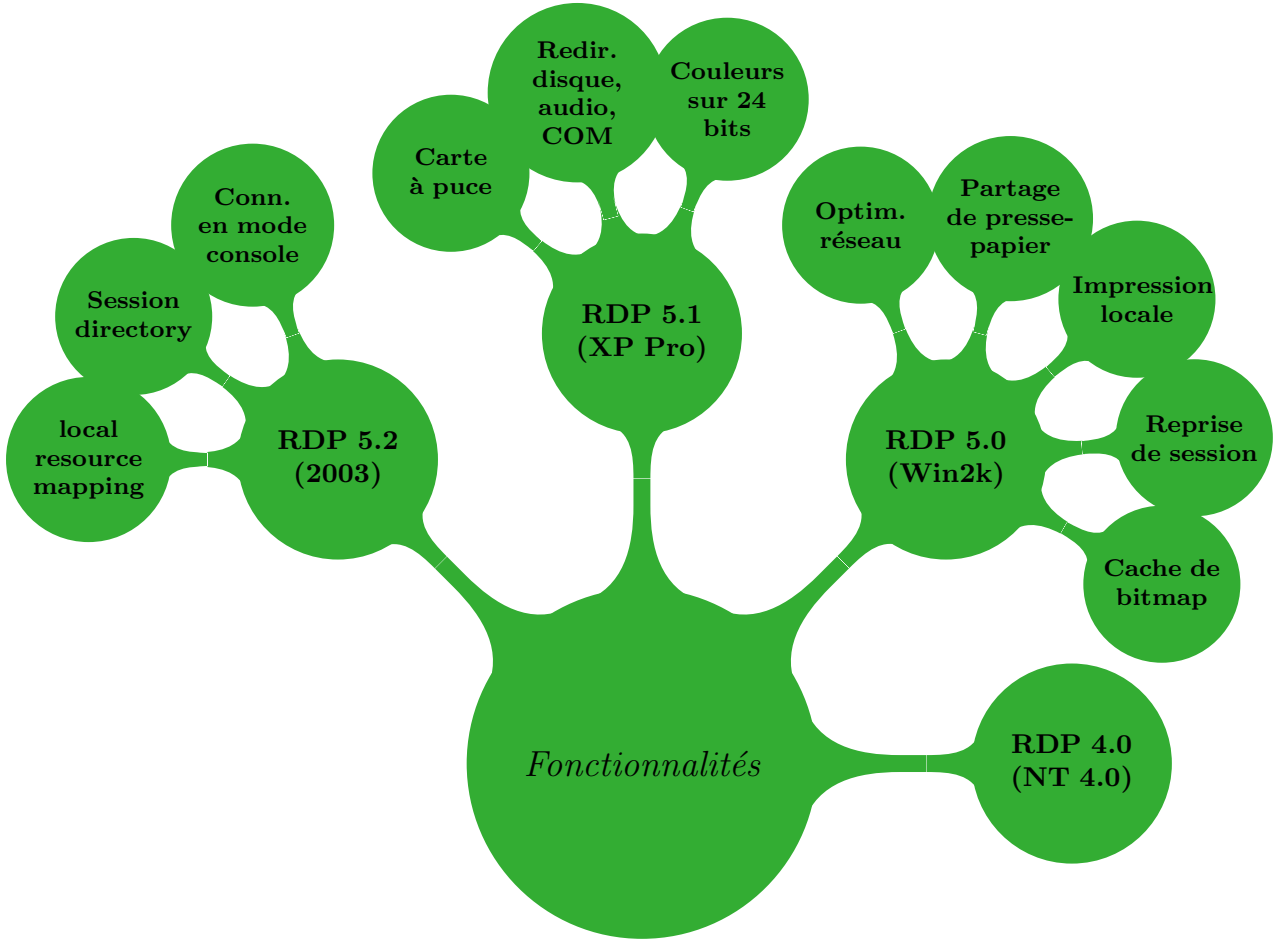
Table des matières

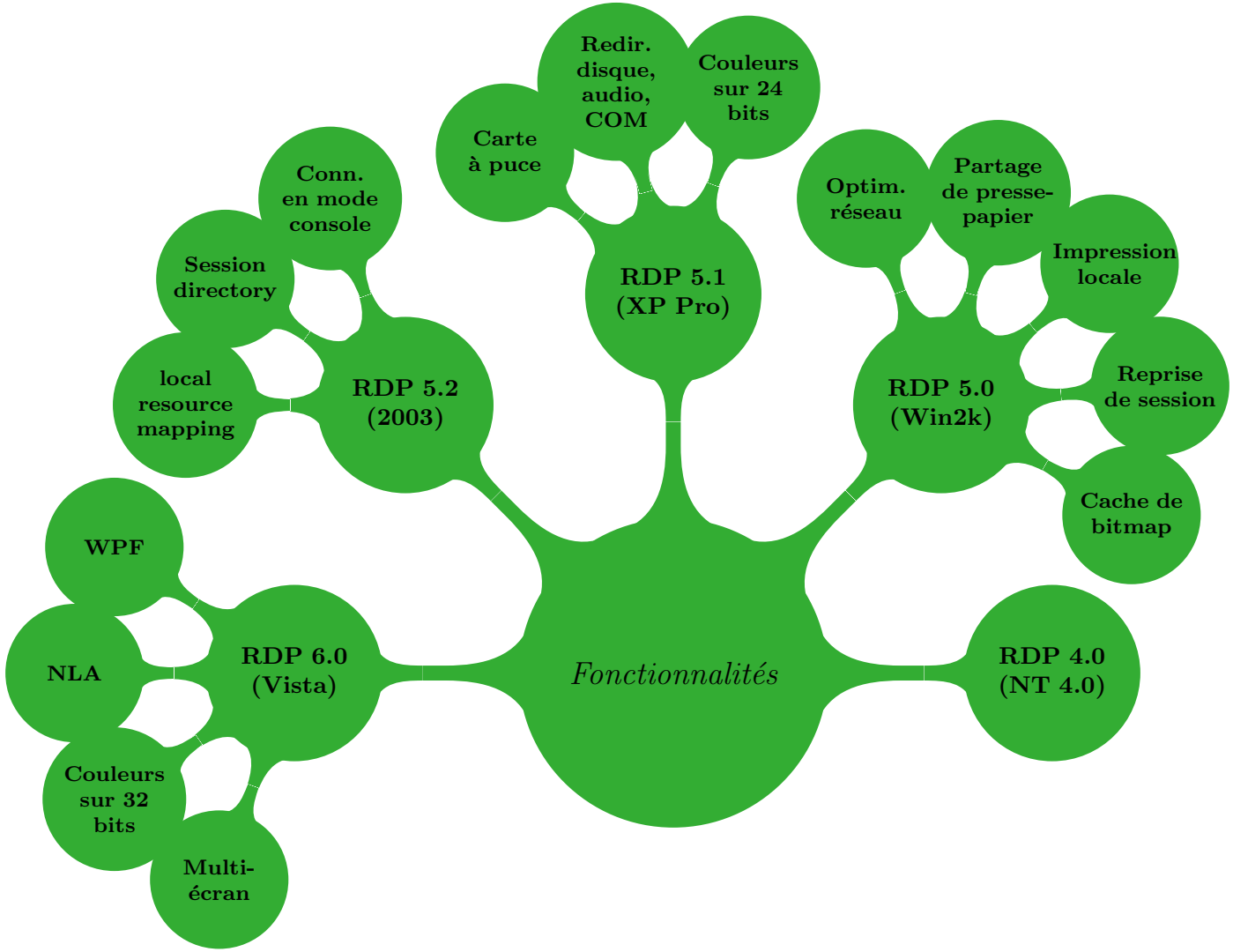

Fonctionnalités

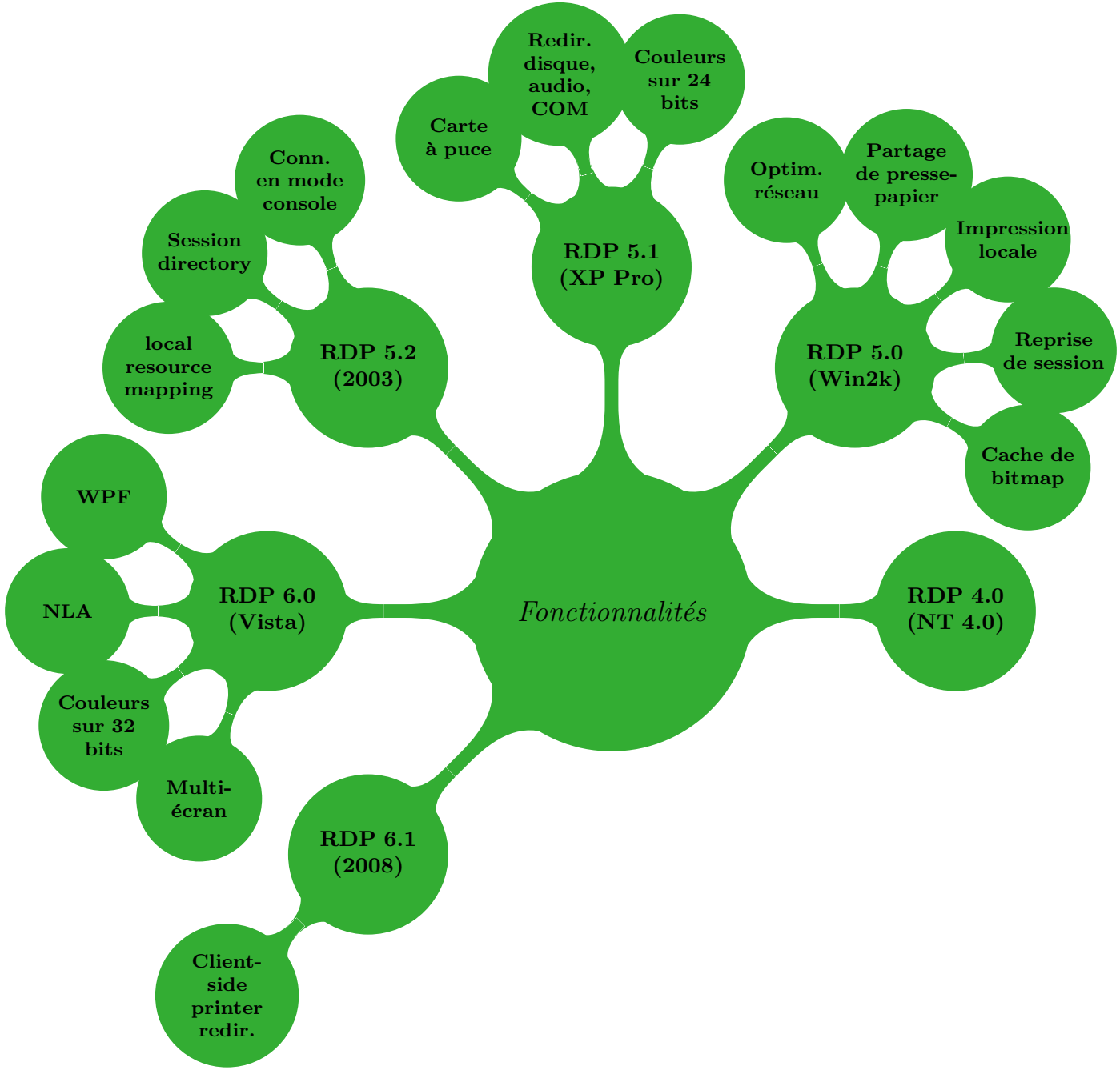

Fonctionnalités
RDP 4.0
(NT 4.0)

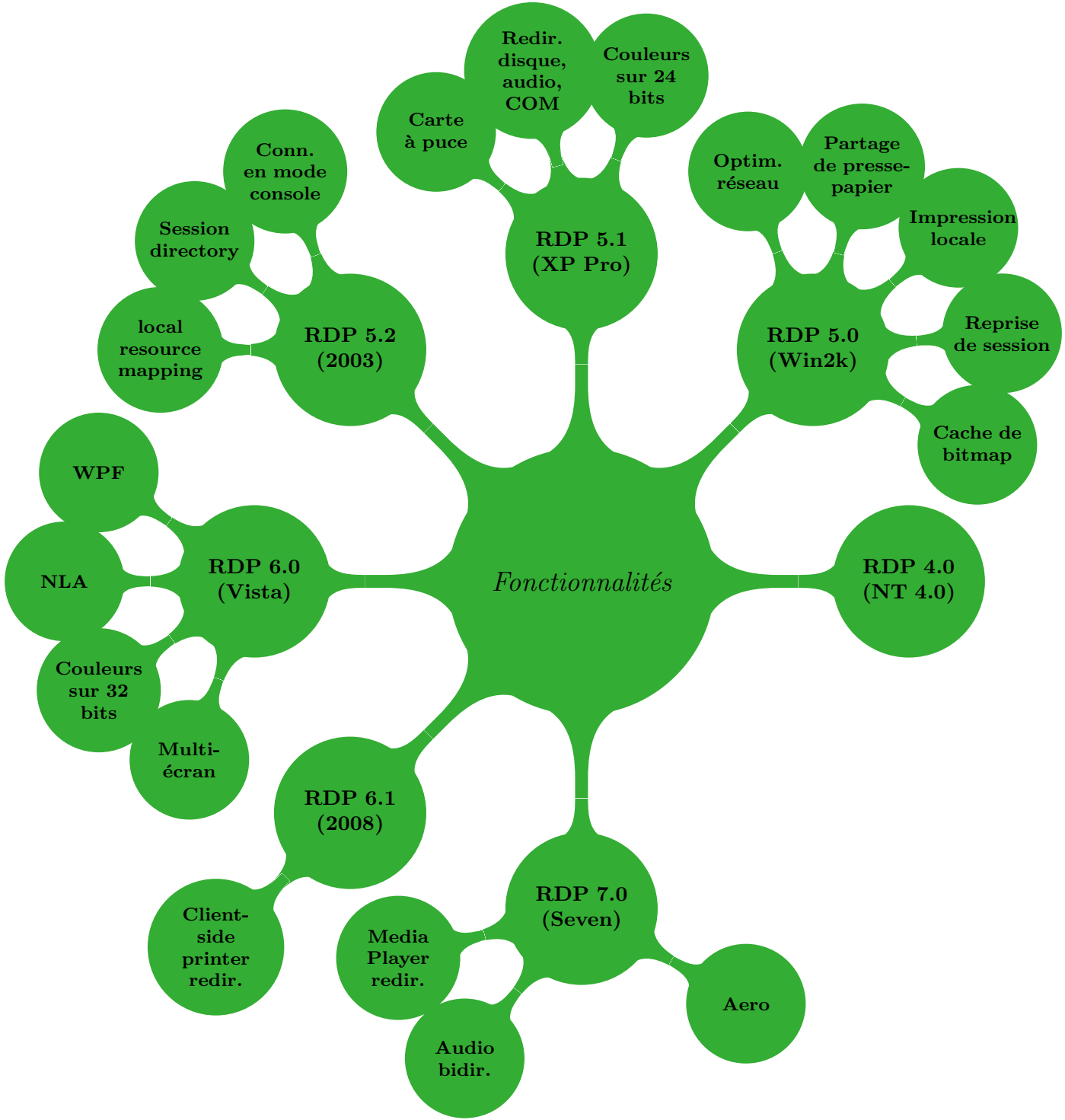

Optim. réseau
Partage de presse-papier
Impression locale
RDP 5.0 (Win2k)
Reprise de session
Cache de bitmap
Fonctionnalités
RDP 4.0 (NT 4.0)


Redir. disque, audio, COM
Couleurs sur 24 bits
Carte à puce
RDP 5.1 (XP Pro)
Optim. réseau
Partage de presse-papier
Impression locale
RDP 5.0 (Win2k)
Reprise de session
Cache de bitmap
Fonctionnalités
RDP 4.0 (NT 4.0)


Fonctionnalités
RDP 4.0 (NT 4.0)
RDP 5.0 (Win2k)
Optim. réseau
Partage de presse-papier
Impression locale
Reprise de session
Cache de bitmap
RDP 5.1 (XP Pro)
Redir. disque, audio, COM
Couleurs sur 24 bits
Carte à puce
RDP 5.2 (2003)
Conn. en mode console
Session directory
local resource mapping


Fonctionnalités
RDP 4.0 (NT 4.0)
RDP 5.0 (Win2k)
Optim. réseau
Partage de presse-papier
Impression locale
Reprise de session
Cache de bitmap
RDP 5.1 (XP Pro)
Carte à puce
Redir. disque, audio, COM
Couleurs sur 24 bits
RDP 5.2 (2003)
Conn. en mode console
Session directory
local resource mapping
RDP 6.0 (Vista)
WPF
NLA
Couleurs sur 32 bits
Multi-écran


Fonctionnalités
RDP 4.0 (NT 4.0)
RDP 5.0 (Win2k)
Optim. réseau
Partage de presse-papier
Impression locale
Reprise de session
Cache de bitmap
RDP 5.1 (XP Pro)
Carte à puce
Redir. disque, audio, COM
Couleurs sur 24 bits
RDP 5.2 (2003)
Conn. en mode console
Session directory
local resource mapping
RDP 6.0 (Vista)
WPF
NLA
Couleurs sur 32 bits
Multi-écran
RDP 6.1 (2008)
Client-side printer redir.


Fonctionnalités
RDP 4.0 (NT 4.0)
RDP 5.0 (Win2k)
Optim. réseau
Partage de presse-papier
Impression locale
Reprise de session
Cache de bitmap
RDP 5.1 (XP Pro)
Redir. disque, audio, COM
Couleurs sur 24 bits
Carte à puce
RDP 5.2 (2003)
Conn. en mode console
Session directory
local resource mapping
RDP 6.0 (Vista)
WPF
NLA
Couleurs sur 32 bits
Multi-écran
RDP 6.1 (2008)
Client-side printer redir.
RDP 7.0 (Seven)
Media Player redir.
Audio bidir.
Aero

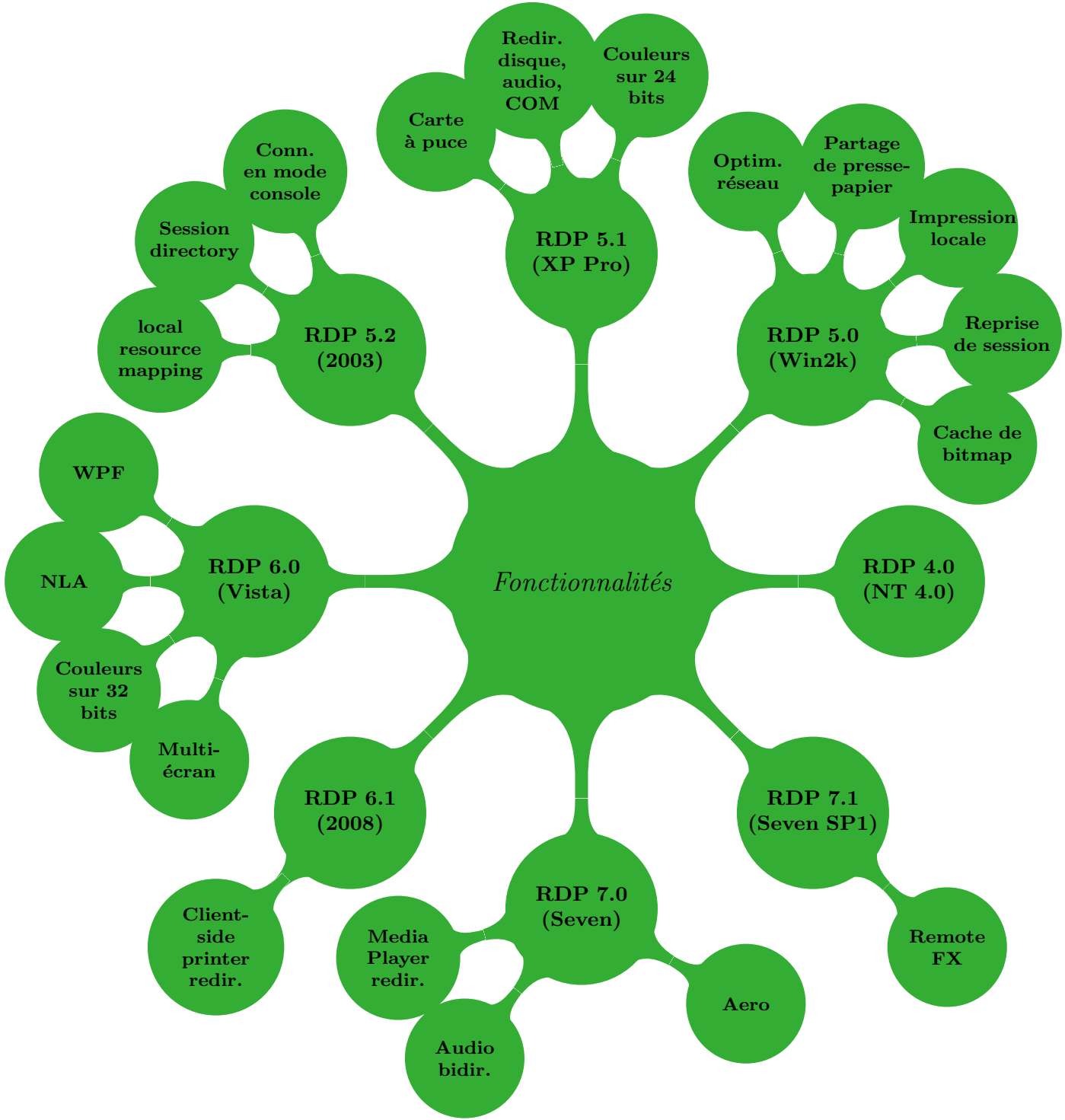
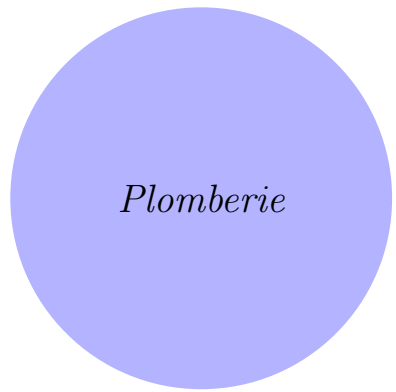
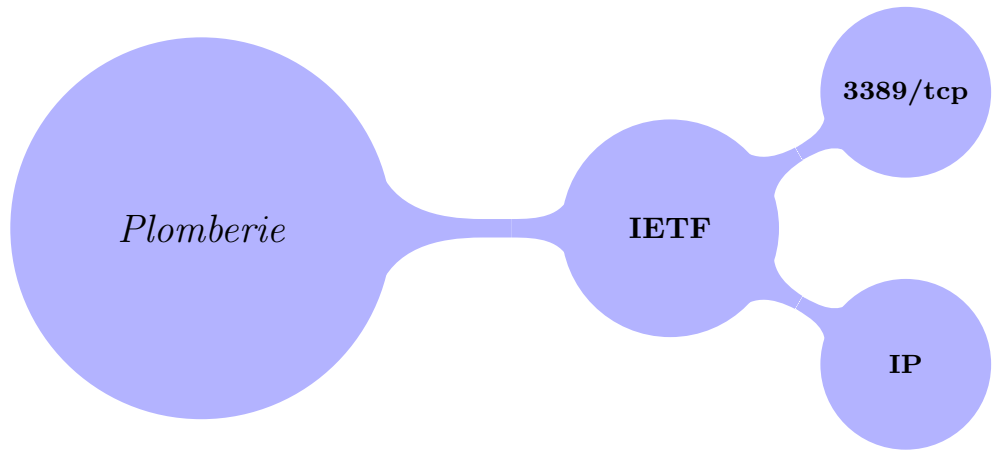
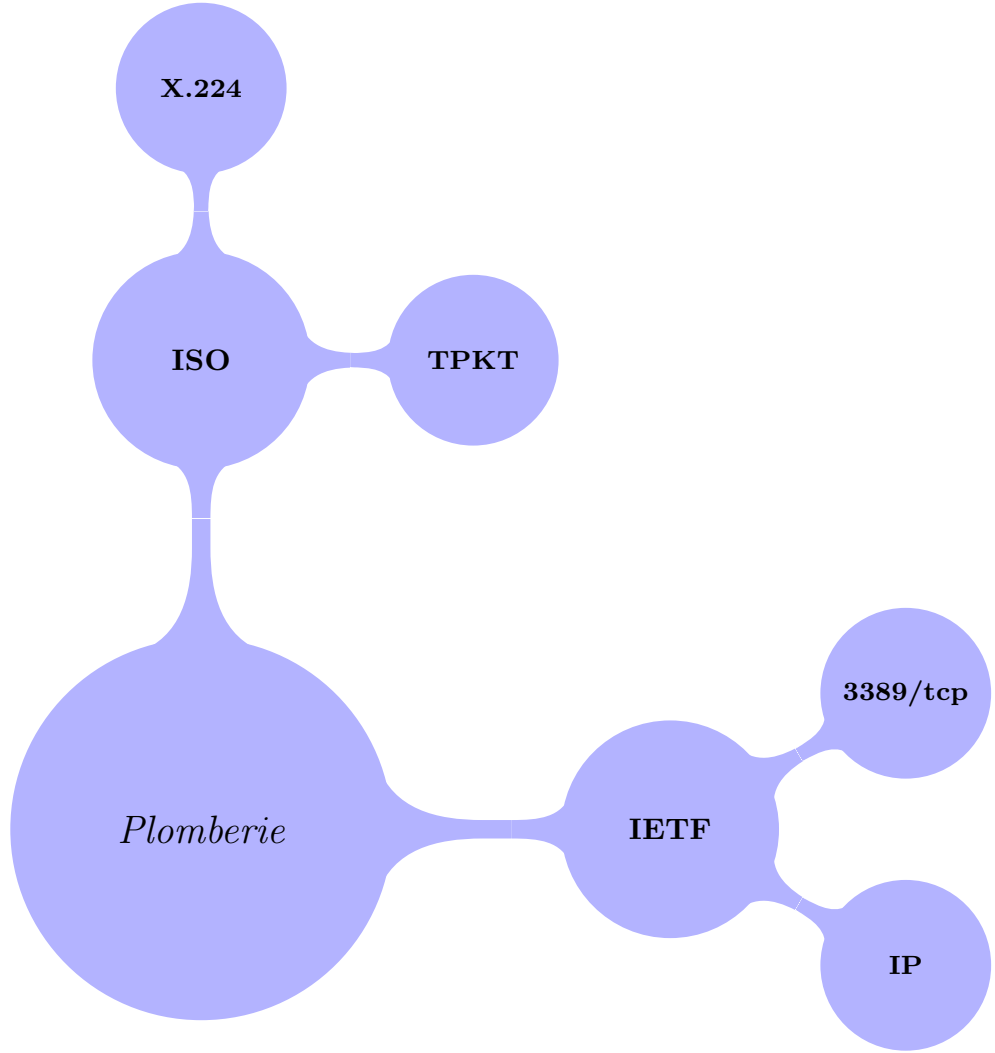
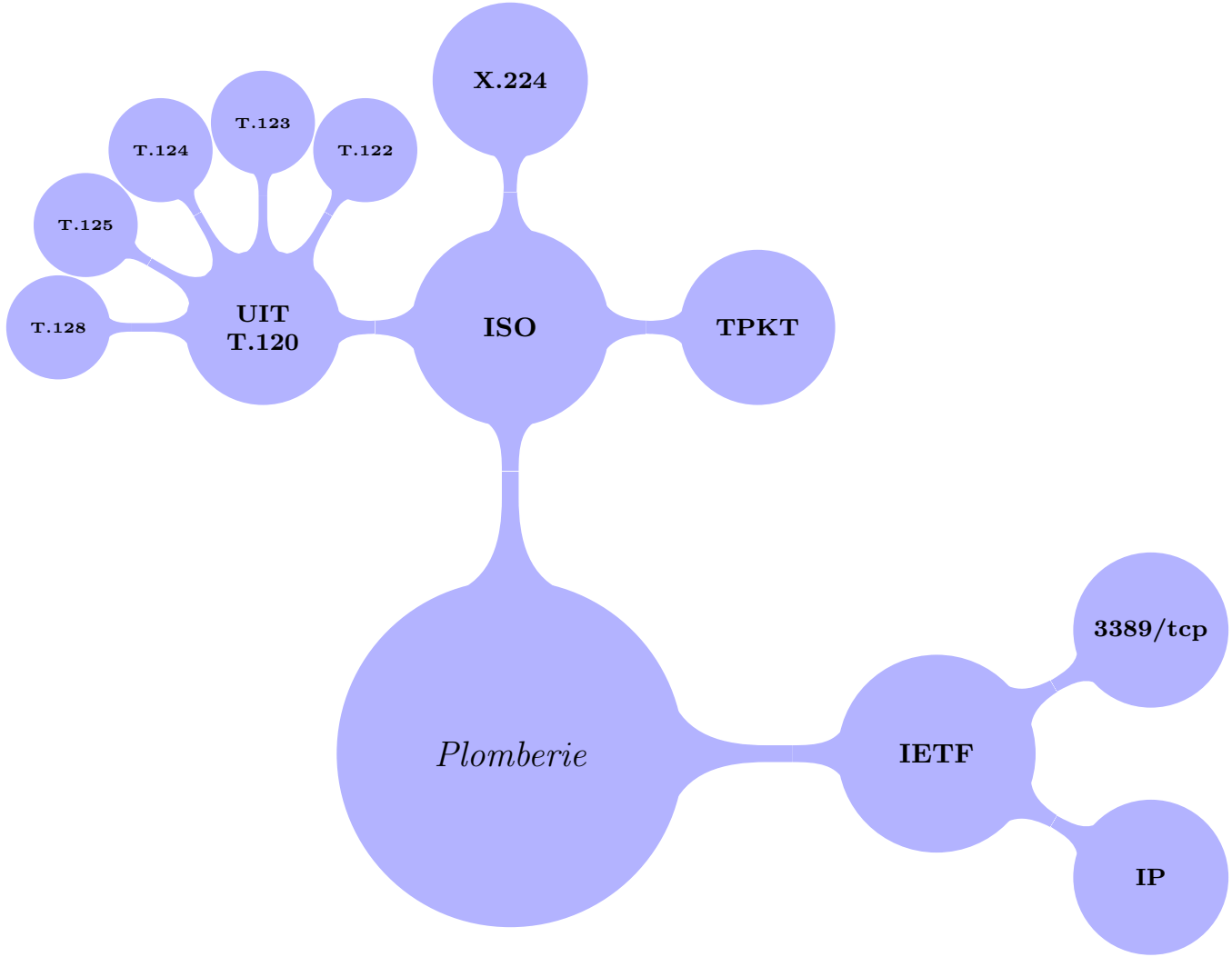

Fonctionnalités
RDP 4.0 (NT 4.0)
RDP 5.0 (Win2k)
Optim. réseau
Partage de presse-papier
Impression locale
Reprise de session
Cache de bitmap
RDP 5.1 (XP Pro)
Redir. disque, audio, COM
Couleurs sur 24 bits
Carte à puce
RDP 5.2 (2003)
Conn. en mode console
Session directory
local resource mapping
RDP 6.0 (Vista)
WPF
NLA
Couleurs sur 32 bits
Multi-écran
RDP 6.1 (2008)
Client-side printer redir.
RDP 7.0 (Seven)
Media Player redir.
Audio bidir.
Aero
RDP 7.1 (Seven SP1)
Remote FX

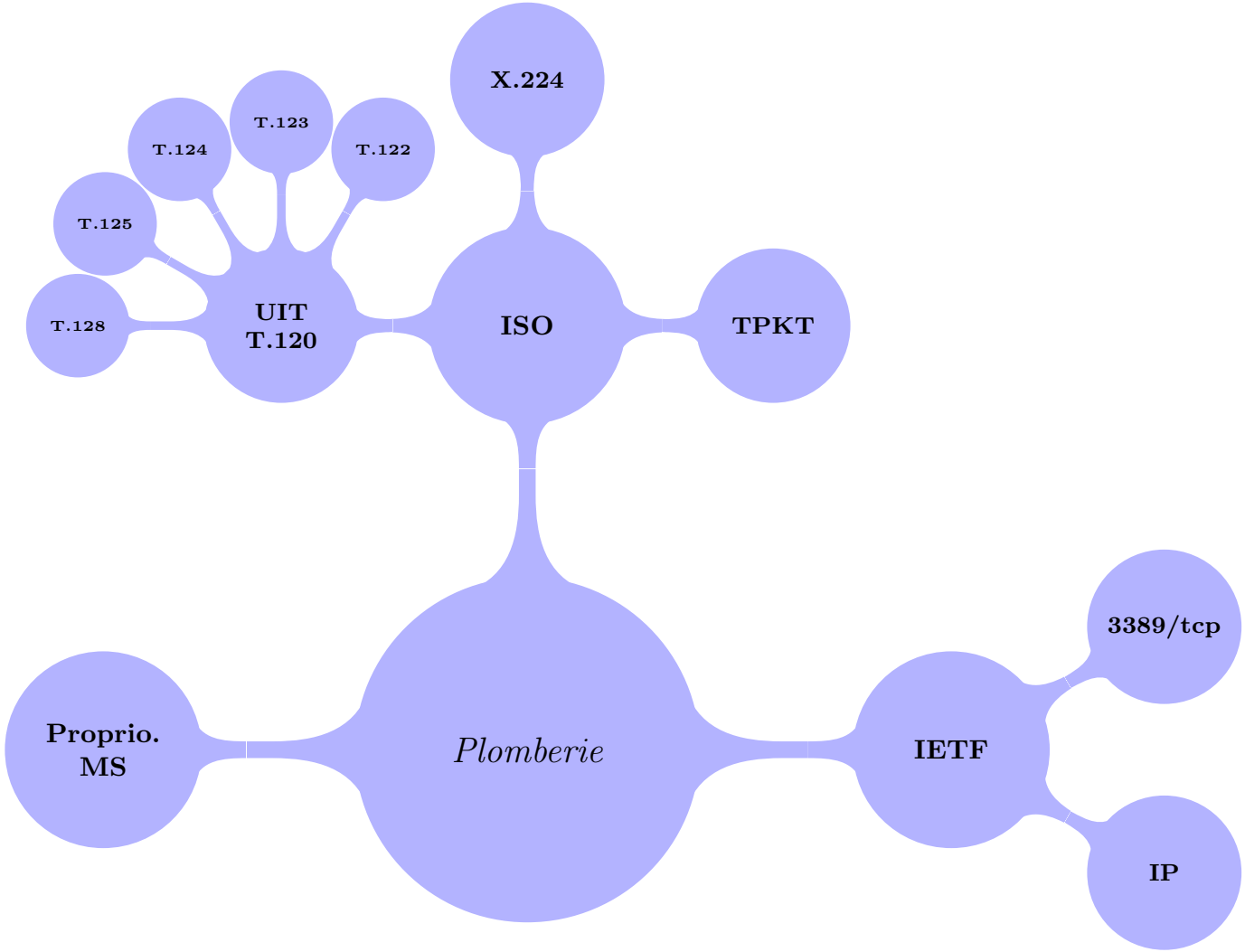
Table des matières

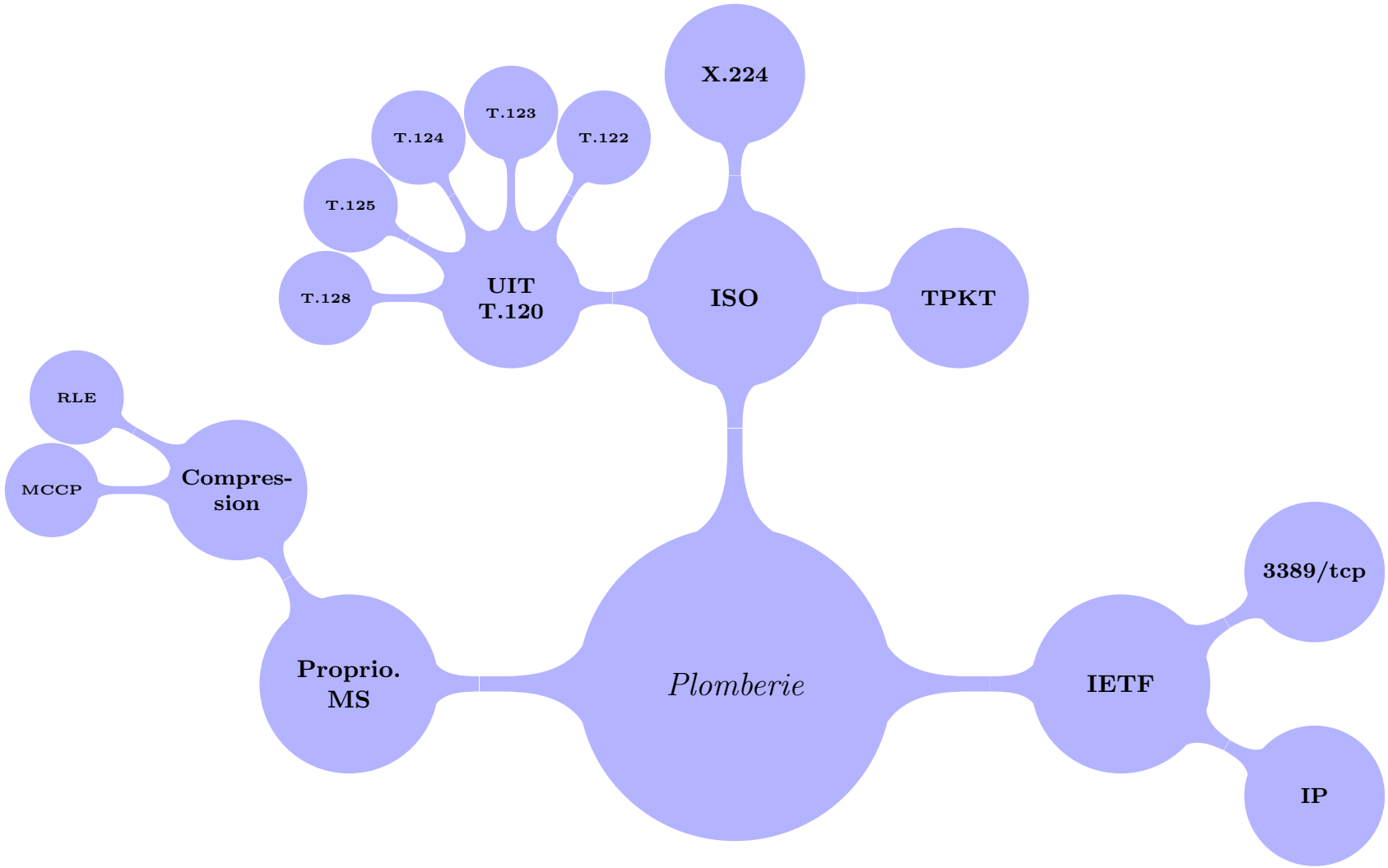

Plomberie

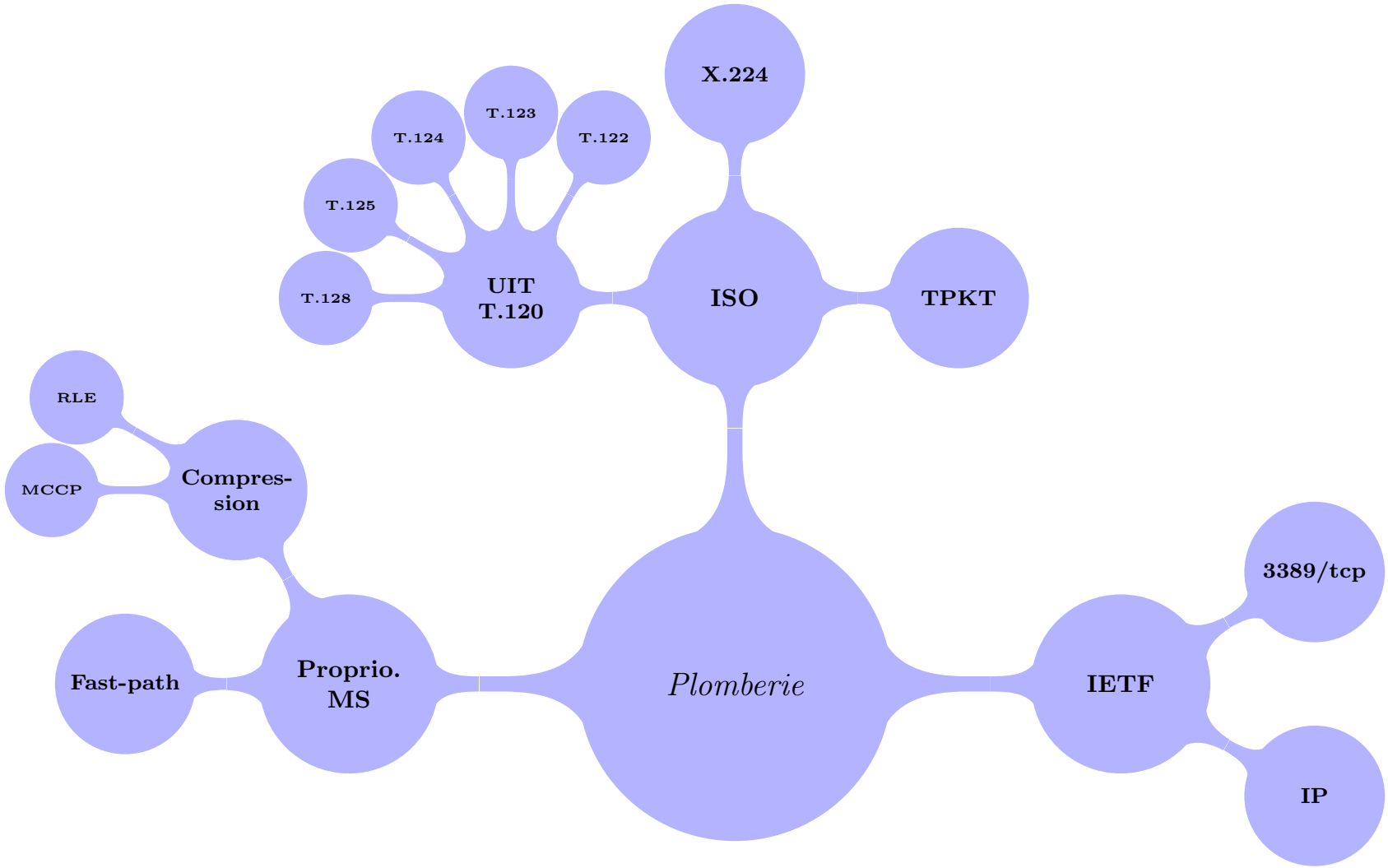

Plomberie
IETF
3389/tcp
IP

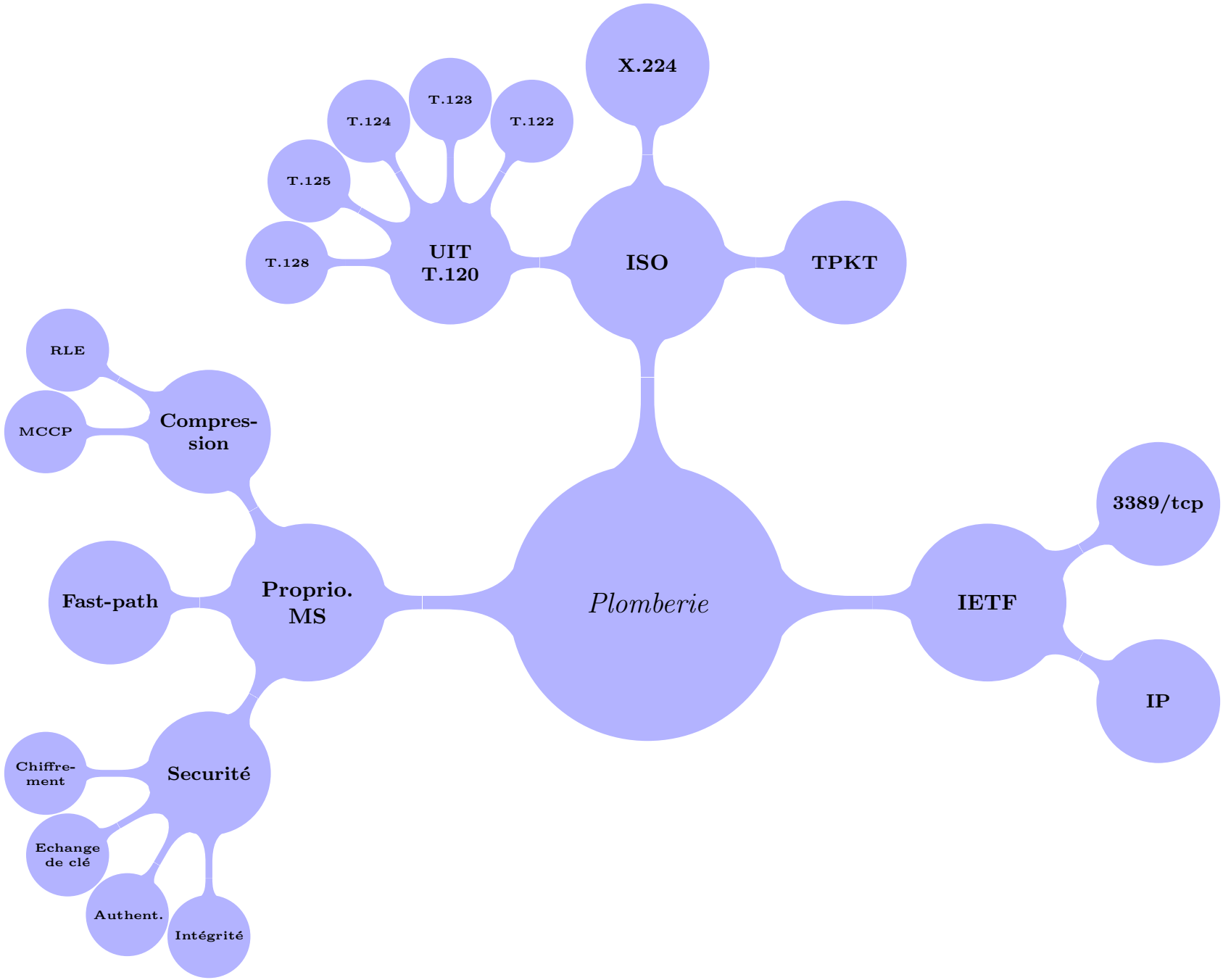

X.224
ISO
TPKT
Plomberie
IETF
3389/tcp
IP

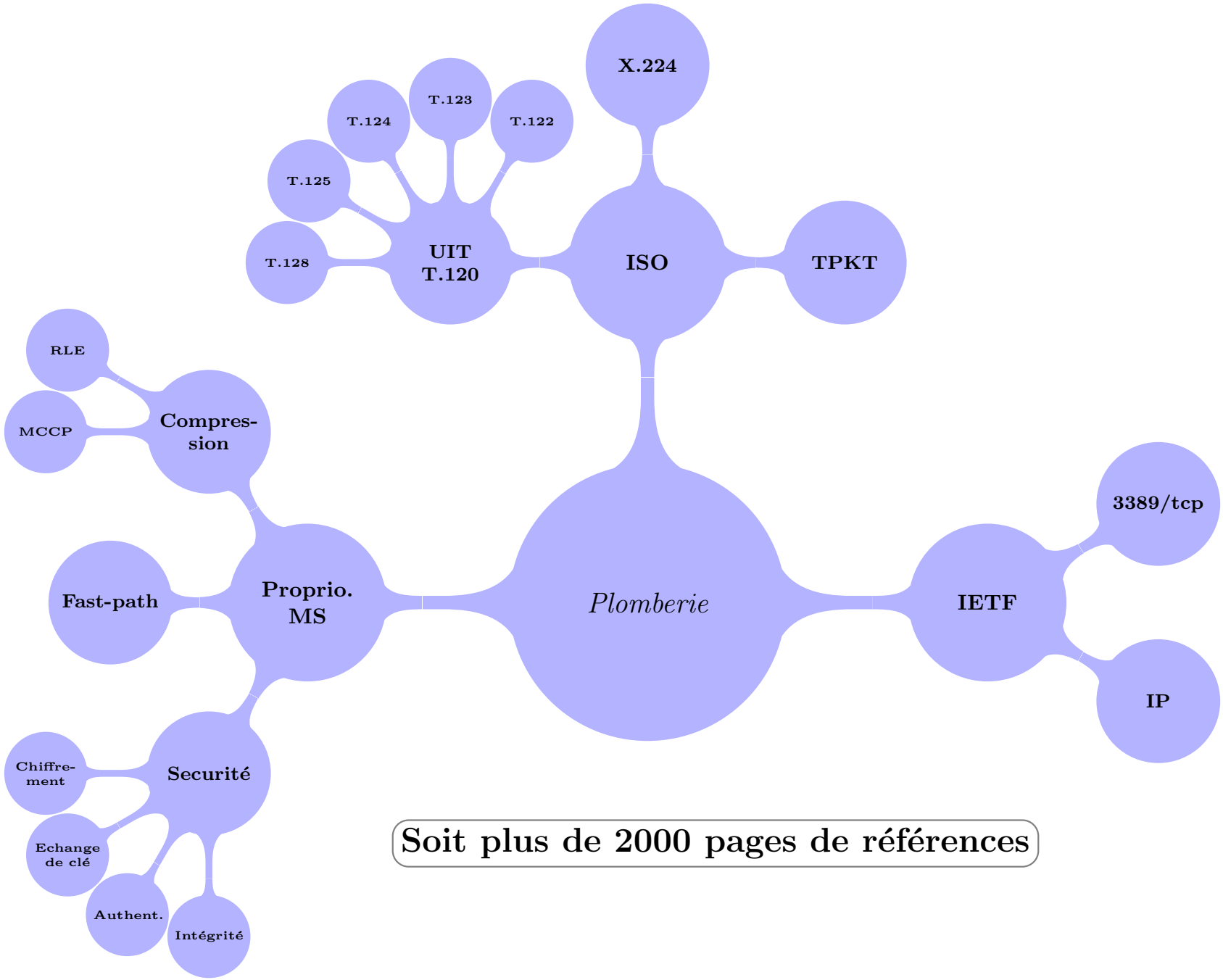

X.224
T.123
T.124
T.122
T.125
T.128
UIT
T.120
ISO
TPKT
Plomberie
IETF
3389/tcp
IP


X.224
T.123
T.124
T.122
T.125
T.128
UIT
T.120
ISO
TPKT
3389/tcp
Proprio.
MS
Plomberie
IETF
IP

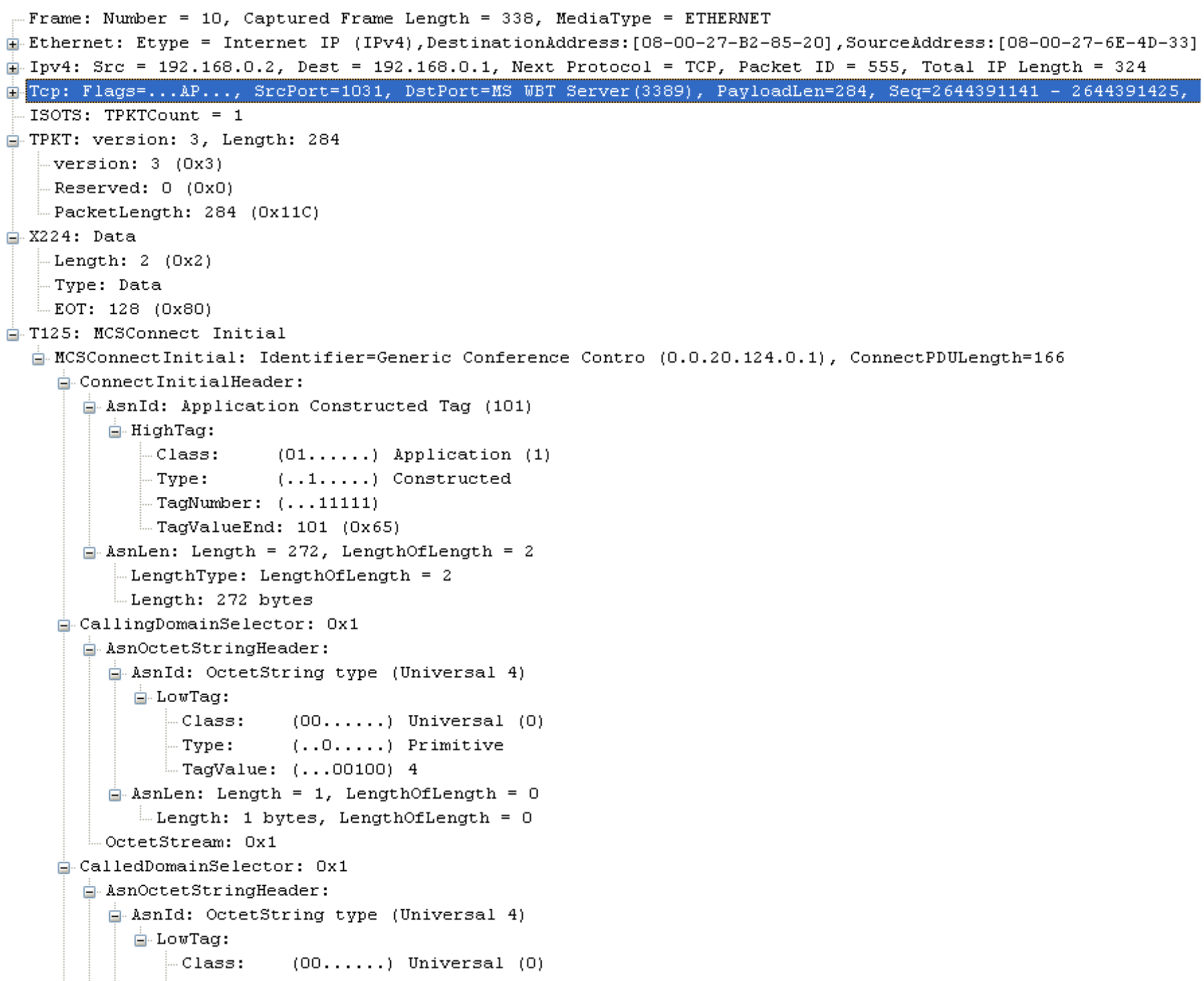

X.224
T.123
T.124
T.122
T.125
T.128
UIT
T.120
ISO
TPKT
RLE
MCCP
Compres-
sion
Proprio.
MS
Plomberie
IETF
3389/tcp
IP


X.224
T.123
T.124
T.122
T.125
T.128
UIT
T.120
ISO
TPKT
RLE
MCCP
Compres-
sion
Fast-path
Proprio.
MS
Plomberie
3389/tcp
IETF
IP


X.224
T.123
T.124
T.122
T.125
T.128
UIT
T.120
ISO
TPKT
RLE
MCCP
Compres-
sion
Fast-path
Proprio.
MS
Plomberie
3389/tcp
IETF
IP
Chiffre-
ment
Securité
Echange
de clé
Authent.
Intégrité



Soit plus de 2000 pages de références


Frame: Number = 10, Captured Frame Length = 338, MediaType = ETHERNET
Ethernet: Etype = Internet IP (IPv4),DestinationAddress:[08-00-27-B2-85-20],SourceAddress:[08-00-27-6E-4D-33]
Ipv4: Src = 192.168.0.2, Dest = 192.168.0.1, Next Protocol = TCP, Packet ID = 555, Total IP Length = 324
Tcp: Flags=...AP..., SrcPort=1031, DstPort=MS WBT Server(3389), PayloadLen=284, Seq=2644391141 - 2644391425,
ISOTS: TPKTCount = 1
TPKT: version: 3, Length: 284
version: 3 (0x3)
Reserved: 0 (0x0)
PacketLength: 284 (0x11C)
X224: Data
Length: 2 (0x2)
Type: Data
EOT: 128 (0x80)
T125: MCSConnect Initial
MCSConnectInitial: Identifier=Generic Conference Contro (0.0.20.124.0.1), ConnectPDULength=166
ConnectInitialHeader:
AsnId: Application Constructed Tag (101)
HighTag:
Class: (01......) Application (1)
Type: (..1.....) Constructed
TagNumber: (...11111)
TagValueEnd: 101 (0x65)
AsnLen: Length = 272, LengthOfLength = 2
LengthType: LengthOfLength = 2
Length: 272 bytes
CallingDomainSelector: 0x1
AsnOctetStringHeader:
AsnId: OctetString type (Universal 4)
LowTag:
Class: (00......) Universal (0)
Type: (..0.....) Primitive
TagValue: (...00100) 4
AsnLen: Length = 1, LengthOfLength = 0
Length: 1 bytes, LengthOfLength = 0
OctetStream: 0x1
CalledDomainSelector: 0x1
AsnOctetStringHeader:
AsnId: OctetString type (Universal 4)
LowTag:
Class: (00......) Universal (0)

```
Frame: Number = 10, Captured Frame Length = 338, MediaType = ETHERNET
Ethernet: Etype = Internet IP (IPv4),DestinationAddress:[08-00-27-B2-85-20],SourceAddress:[08-00-27-6E-4D-33]
Ipv4: Src = 192.168.0.2, Dest = 192.168.0.1, Next Protocol = TCP, Packet ID = 555, Total IP Length = 324
Tcp: Flags=...AP..., SrcPort=1031, DstPort=MS WBT Server(3389), PayloadLen=284, Seq=2644391141 - 2644391425,
ISOTS: TPKTCount = 1
TPKT: version: 3, Length: 284
  version: 3 (0x3)
  Reserved: 0 (0x0)
  PacketLength: 284 (0x11C)
X224: Data
  Length: 2 (0x2)
  Type: Data
  EOT: 128 (0x80)
T125: MCSConnect Initial
- MCSConnectInitial: Identifier=Generic Conference Contro (0.0.20.124.0.1), ConnectPDULength=166
  - ConnectInitialHeader:
    - AsnId: Application Constructed Tag (101)
      - HighTag:
          Class:     (01......) Application (1)
          Type:      (..1.....) Constructed
          TagNumber: (...11111)
          TagValueEnd: 101 (0x65)
    - AsnLen: Length = 272, LengthOfLength = 2
        LengthType: LengthOfLength = 2
        Length: 272 bytes
  - CallingDomainSelector: 0x1
    - AsnOctetStringHeader:
      - AsnId: OctetString type (Universal 4)
        - LowTag:
            Class:     (00......) Universal (0)
            Type:      (..0.....) Primitive
            TagValue: (...00100) 4
      - AsnLen: Length = 1, LengthOfLength = 0
          Length: 1 bytes, LengthOfLength = 0
      OctetStream: 0x1
  - CalledDomainSelector: 0x1
    - AsnOctetStringHeader:
      - AsnId: OctetString type (Universal 4)
        - LowTag:
            Class:     (00......) Universal (0)
            Type:      (..0.....) Primitive
            TagValue: (...00100) 4
      - AsnLen: Length = 1, LengthOfLength = 0
          Length: 1 bytes, LengthOfLength = 0
      OctetStream: 0x1
  - UpwardFlag: True
    - AsnBooleanHeader:
      - AsnId: Boolean type (Universal 1)
        - LowTag:
            Class:     (00......) Universal (0)
            Type:      (..0.....) Primitive
            TagValue: (...00001) 1
      - AsnLen: Length = 1, LengthOfLength = 0
          Length: 1 bytes, LengthOfLength = 0
      Tf: 255 (0xFF)
  - TargetParameters: Length = 25, LengthOfLength = 0
    - DomainParametersHeader: 0x1
      - AsnId: Sequence and SequenceOf types (Universal 16)
        - LowTag:
            Class:     (00......) Universal (0)
            Type:      (..1.....) Constructed
            TagValue: (...10000) 16
      - AsnLen: Length = 25, LengthOfLength = 0
          Length: 25 bytes, LengthOfLength = 0
    - ChannelIds: 34
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 34 (0x22)
    - UserIDs: 2
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 2 (0x2)
    - TokenIds: 0
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 0 (0x0)
    - NumPriorities: 1
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 1 (0x1)
    - MinThroughput: 0
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 0 (0x0)
    - Height: 1
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 1 (0x1)
    - MCSPDUsize: 65535
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 2, LengthOfLength = 0
            Length: 2 bytes, LengthOfLength = 0
        AsnInt: 65535 (0xFFFF)
    - protocolVersion: 2
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 2 (0x2)
  - MinimumParameters: Length = 25, LengthOfLength = 0
    - DomainParametersHeader: 0x1
      - AsnId: Sequence and SequenceOf types (Universal 16)
        - LowTag:
            Class:     (00......) Universal (0)
            Type:      (..1.....) Constructed
            TagValue: (...10000) 16
      - AsnLen: Length = 25, LengthOfLength = 0
          Length: 25 bytes, LengthOfLength = 0
    - ChannelIds: 1
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 1 (0x1)
    - UserIDs: 1
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 1 (0x1)
    - TokenIds: 1
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 1 (0x1)
    - NumPriorities: 1
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 1 (0x1)
    - MinThroughput: 0
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 0 (0x0)
    - Height: 1
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 1 (0x1)
    - MCSPDUsize: 1056
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 2, LengthOfLength = 0
            Length: 2 bytes, LengthOfLength = 0
        AsnInt: 1056 (0x420)
    - protocolVersion: 2
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 2 (0x2)
  - MaximumParameters: Length = 28, LengthOfLength = 0
    - DomainParametersHeader: 0x1
      - AsnId: Sequence and SequenceOf types (Universal 16)
        - LowTag:
            Class:     (00......) Universal (0)
            Type:      (..1.....) Constructed
            TagValue: (...10000) 16
      - AsnLen: Length = 28, LengthOfLength = 0
          Length: 28 bytes, LengthOfLength = 0
    - ChannelIds: 65535
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 2, LengthOfLength = 0
            Length: 2 bytes, LengthOfLength = 0
        AsnInt: 65535 (0xFFFF)
    - UserIDs: 64535
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          + LowTag:
        - AsnLen: Length = 2, LengthOfLength = 0
            Length: 2 bytes, LengthOfLength = 0
        AsnInt: 64535 (0xFC17)
    - TokenIds: 65535
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 2, LengthOfLength = 0
            Length: 2 bytes, LengthOfLength = 0
        AsnInt: 65535 (0xFFFF)
    - NumPriorities: 1
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 1 (0x1)
    - MinThroughput: 0
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 0 (0x0)
    - Height: 1
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 1 (0x1)
    - MCSPDUsize: 65535
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 2, LengthOfLength = 0
            Length: 2 bytes, LengthOfLength = 0
        AsnInt: 65535 (0xFFFF)
    - protocolVersion: 2
      - AsnIntegerHeader:
        - AsnId: Integer type (Universal 2)
          - LowTag:
              Class:     (00......) Universal (0)
              Type:      (..0.....) Primitive
              TagValue: (...00010) 2
        - AsnLen: Length = 1, LengthOfLength = 0
            Length: 1 bytes, LengthOfLength = 0
        AsnInt: 2 (0x2)
- UserData: Identifier=Generic Conference Contro (0.0.20.124.0.1), ConnectPDULength=166
  - UserDataHeader:
    - AsnId: OctetString type (Universal 4)
      - LowTag:
          Class:     (00......) Universal (0)
          Type:      (..0.....) Primitive
          TagValue: (...00100) 4
    - AsnLen: Length = 175, LengthOfLength = 2
        LengthType: LengthOfLength = 2
        Length: 175 bytes
  - AsnBerObjectIdentifier: Generic Conference Contro (0.0.20.124.0.1)
    - AsnObjectIdentifierHeader:
      - AsnId: Reserved for use by the encoding rules (Universal 0)
        - LowTag:
            Class:     (00......) Universal (0)
            Type:      (..0.....) Primitive
            TagValue: (...00000) 0
      - AsnLen: Length = 5, LengthOfLength = 0
          Length: 5 bytes, LengthOfLength = 0
    - First: 0 (0x0)
    - Final: 20 (0x14)
    - Final: 124 (0x7C)
    - Final: 0 (0x0)
    - Final: 1 (0x1)
  - ConnectPDULength: 166
    - Align: No Padding
    - Length: 166
  - ConnectGCCPDU: conferenceCreateRequest
    - ExtensionBit: 0 (0x0)
    - ChoiceValue: conferenceCreateRequest
      - Value: (000.....) 0x0
    - conferenceCreateRequest:
      - ExtensionBit: 0 (0x0)
      - convenerPasswordPresent: 0 (0x0)
      - passwordPresent: 0 (0x0)
      - conductorPrivilegesPresent: 0 (0x0)
      - conductedPrivilegesPresent: 0 (0x0)
      - nonConductedPrivilegesPresent: 0 (0x0)
      - conferenceDescriptionPresent: 0 (0x0)
      - callerIdentifierPresent: 0 (0x0)
      - userDataPresent: 1 (0x1)
      - conferenceName:
        - ExtensionBit: 0 (0x0)
        - textPresent: 0 (0x0)
        - numeric: 1
          - SimpleNumericString: 1
            - NumericString: 1
              - Align: No Padding
                - Padding1: (0.......) 0x0
              - Length: 1
                - Value: (00000000) 0x0
              - RestrictedstrI: 1
                - FourBits: (0001....) 0x1
      - lockedConference: False
        - Value: False 0.......
      - listedConference: False
        - Value: False 0.......
      - conductibleConference: False
        - Value: False 0.......
      - TerminationMethod: automatic
        - ExtensionBit: 0 (0x0)
        - RootIndex: 0
          - Value: (0.......) 0x0
      - userData:
        - Size: 1
          - Align: No Padding
            - Padding7: (0000000.) 0x0
          - Length: 1
        - UserData: 0x44756361
          - valuePresent: 1 (0x1)
          - Key: h221NonStandard
            - ChoiceValue: h221NonStandard
              - Value: (1.......) 0x1
            - h221NonStandard:
              - H221NonStandardIdentifier: length: 4
                - ConstrainedLength: 4
                  - Value: (00000000) 0x0
                - Align: No Padding
                  - Padding6: (000000..) 0x0
                - Value: Binary Large Object (4 Bytes)
          - ClientMcsConnectInitialPdu:
            - RDPGCCUserDataRequestLength: 152
              - Align: No Padding
              - Length: 152
            - TsUd: CS_CORE
              - TsUdHeader: Type = CS_CORE, Length = 140
                - Type: CS_CORE
                - Length: 140 (0x8C)
              - TsUdCsCore:
                - Version: RDP 5.0, 5.1, 5.2, 6.0, 6.1, and 7.0
                - DesktopWidth: 800 (0x320)
                - DesktopHeight: 600 (0x258)
                - ColorDepth: 8 bpp
                - SASSequence: 0xaa03, SHOULD be set to RNS_UD_SAS_DEL(0xAA03)
                - KeyboardLayout: Language: French, Location: France
                - ClientBuild: 2195 (0x893)
                - ClientName: AWESOME
                - KeyboardType: IBM enhanced (101- or 102-key) keyboard
                - KeyboardSubType: 0 (0x0)
                - KeyboardFunctionKey: 12 (0xC)
                - ImeFileName:
                - PostBeta2ColorDepth: 8 bpp
                - ClientProductId: 0x1, SHOULD be set to initialized to 1
                - SerialNumber: 0x0, SHOULD be set to 0
              - TsUd: CS_SECURITY
                - TsUdHeader: Type = CS_SECURITY, Length = 12
                  - Type: CS_SECURITY
                  - Length: 12 (0xC)
                - TsUdCsSec:
                  - EncryptionMethod:
                    - Support40Bit:  (.............................0) Not Support
                    - Support128Bit: (............................0.) Not Support 128-bit
                    - Reserved1:     (...........................0..)
                    - Support56Bit:  (..........................0...) Not Support 56-bit
                    - SupportFIPS:   (.........................0....) Not Support FIPS Compliant
                    - Reserved2:     (00000000000000000000000000.....)
                  - ExtEncryptionMethod: Support 40-bit;Support 56-bit;
                    - Support40Bit:  (.............................1) Support 40-bit
                    - Support128Bit: (............................0.) Not Support 128-bit
                    - Reserved1:     (...........................0..)
                    - Support56Bit:  (..........................1...) Support 56-bit
                    - SupportFIPS:   (.........................0....) Not Support FIPS Compliant
                    - Reserved2:     (00000000000000000000000000.....)
```

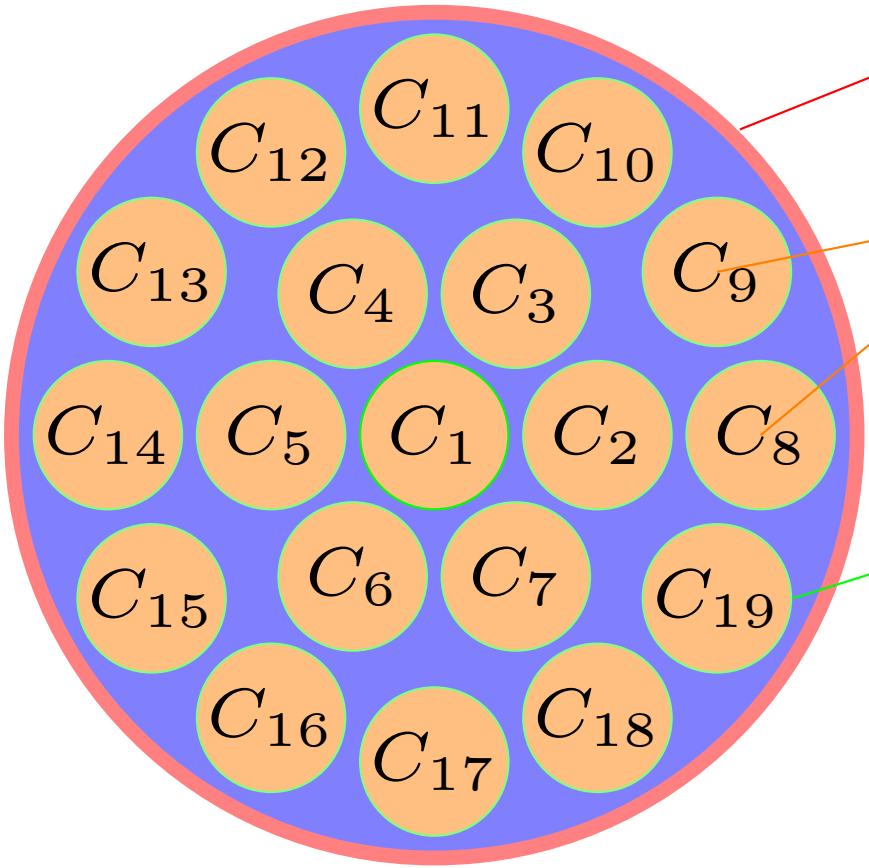
Architecture protocolaire

Panorama :

- RDP procède à la mise en place de canaux de communication (*virtual channels*) entre composants (matériel ou logiciel) des systèmes locaux et distants.
- Ces canaux permettent l'échange de données :
  - Entrées utilisateur clavier/souris (*input events*) : keycodes, etc. ;
  - retour graphiques (*output events*) : bitmap, glyphes, etc. ;
  - copier/coller par le presse-papier, etc. ;
  - montage de système de fichiers ;
  - son ;
  - ...

[Complexité d'une] montée de session RDP :

- 8 étapes distinctes ;
- plusieurs dizaines de paquets échangés ;
- des centaines de paramètres négociés.



Sécurité (C+I)

Canaux virtuels

Compression

Couches de sécurité

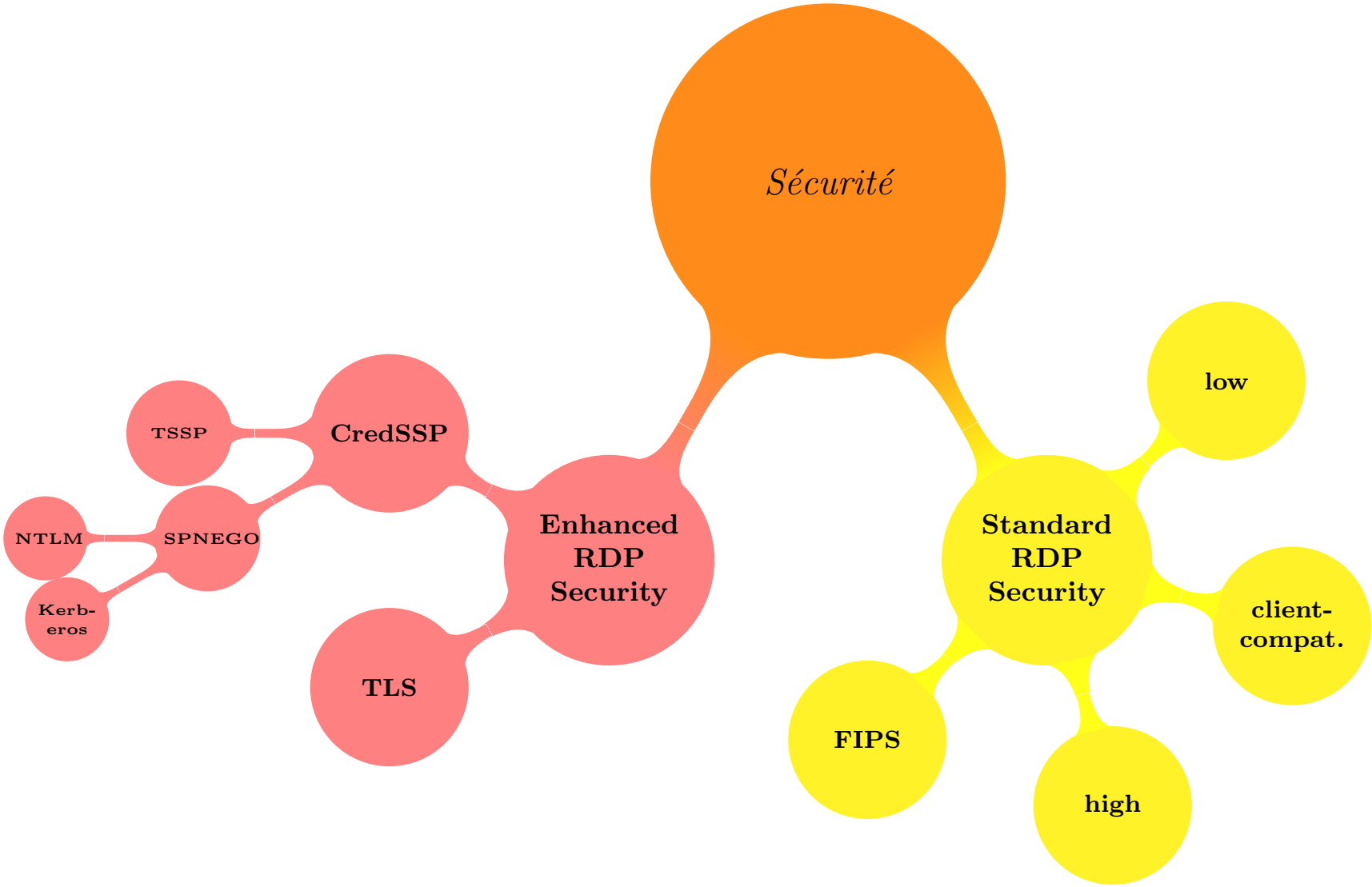


Table des matières

Interlude récréatif #1

Vidéo

Mécanismes *Standard Security* (1/2)

- **Échange de clé**
  - aléa du client chiffré par la clé publique du serveur :
    - clé de 512 bits jusqu'à Windows 2003,
    - 2048 bits depuis Windows 2008 ;
  - pas de *perfect forward secrecy*.

- **Authentification du serveur**
  - initialement inexistante ;
  - puis clé publique signée par une clé privée **documentée** . . .
  - . . . donc inutile.

Mécanismes *Standard Security* (2/2)

- **Intégrité**
  - jusqu'à 5.1 inclus : simple MAC sur les données **en clair** ;
  - à partir de 5.2 : MAC sur les données en clair, avec un compteur.

- **Chiffrement**
  - RC4 :
    - 40, 56 ou 128 bits,
    - logique de choix de la taille complexe,
    - par défaut : taille choisie par le client ;
  - FIPS : triple DES.

- **Conclusion**
  - mécanismes de sécurité *propriétaires* ;
  - nécessité d'évolution.

Table des matières

Mécanismes *Enhanced Security*

TLS :

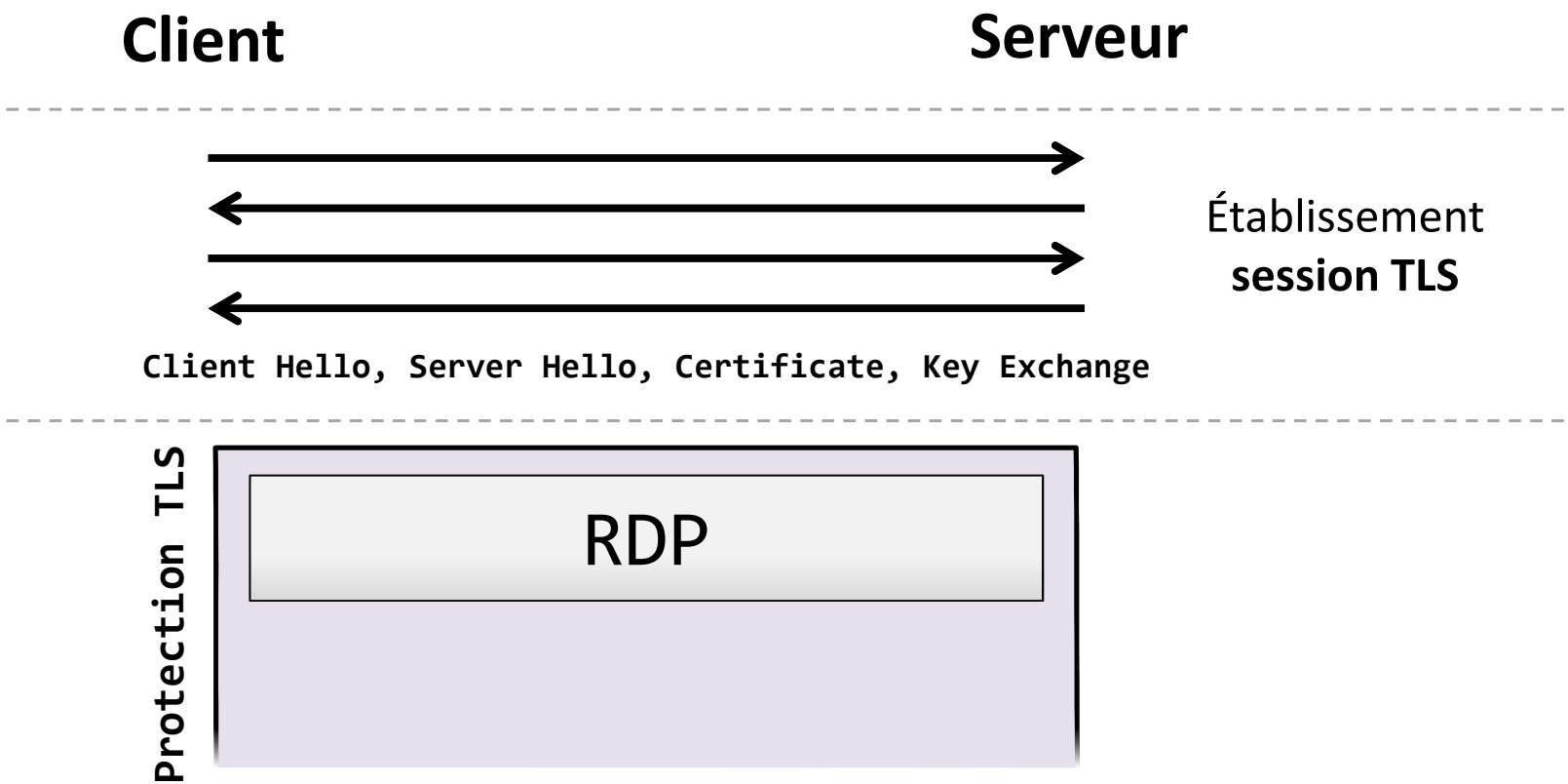
- introduit Windows 2003 SP1 ;
- permet l'authentification du serveur ;
- authentification TLS par certificat client non supportée.

NLA [1]/CredSSP :

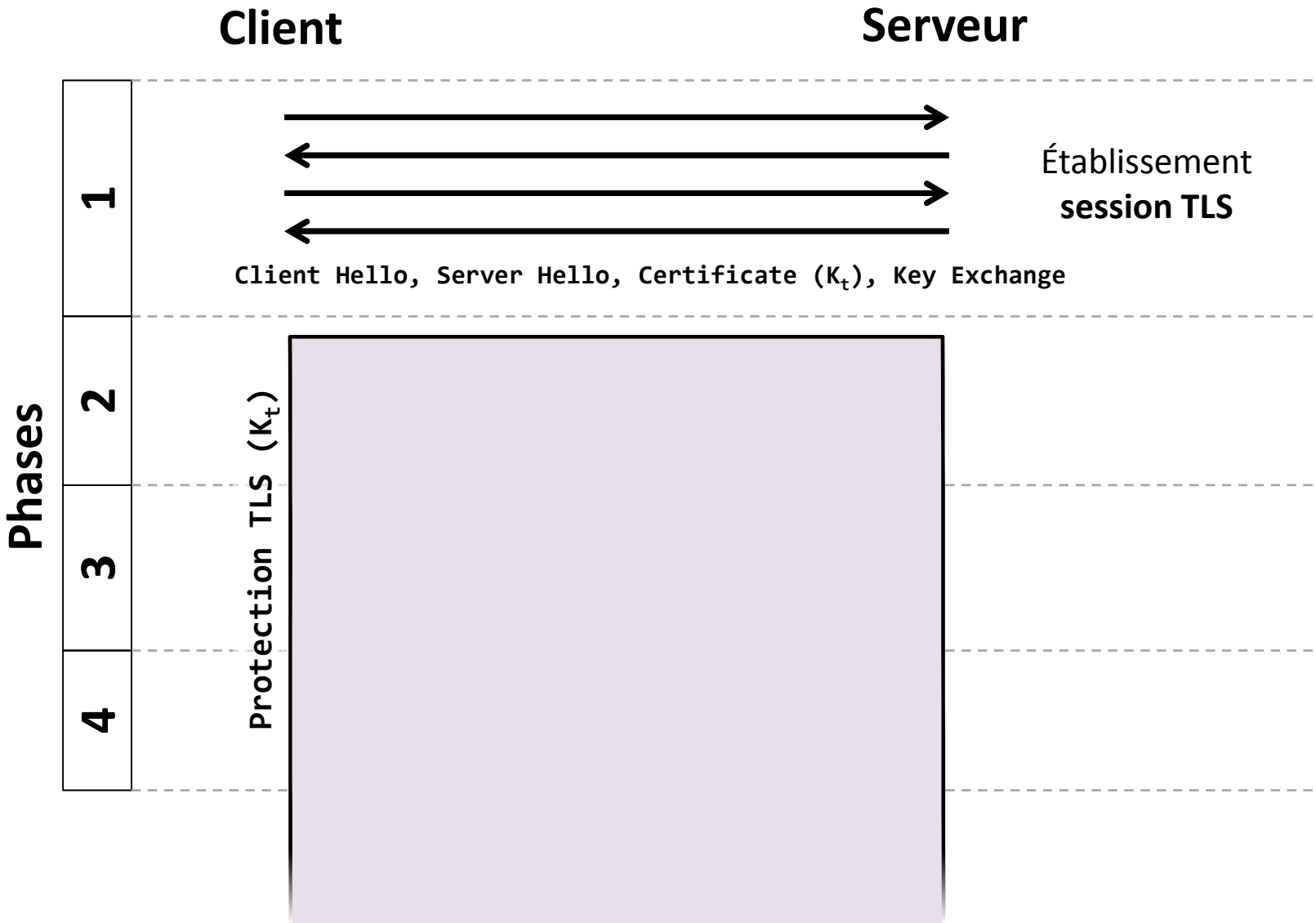
- introduit avec Windows Vista/2008 ;
- repose toujours sur TLS pour la protection des échanges ;
- intègre l'authentification au protocole ;
- permet la délégation des authentifiants au serveur ;
- permet l'authentification du serveur par Kerberos.

1. *Network Level Authentication*

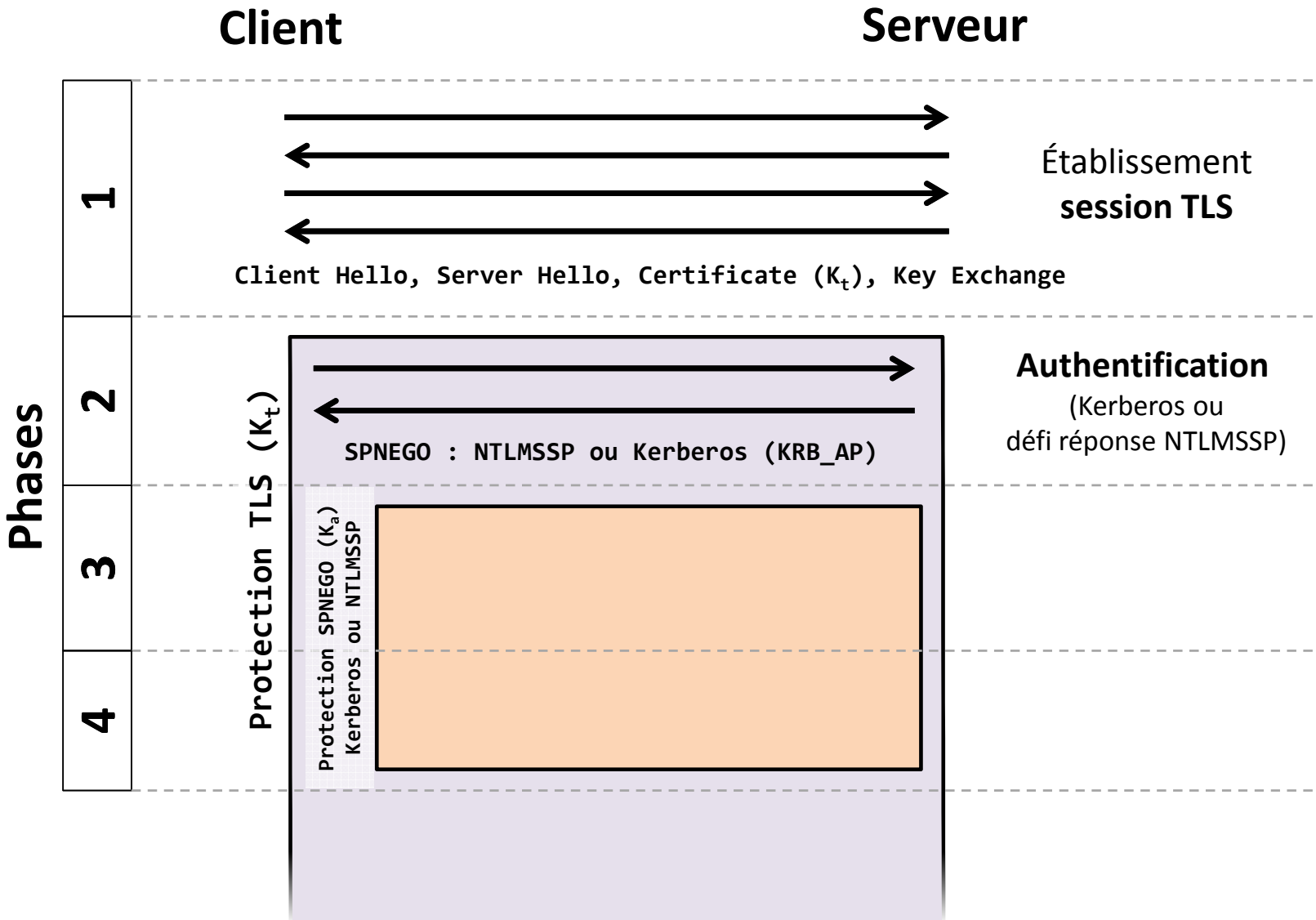
Couches protocolaires avec TLS



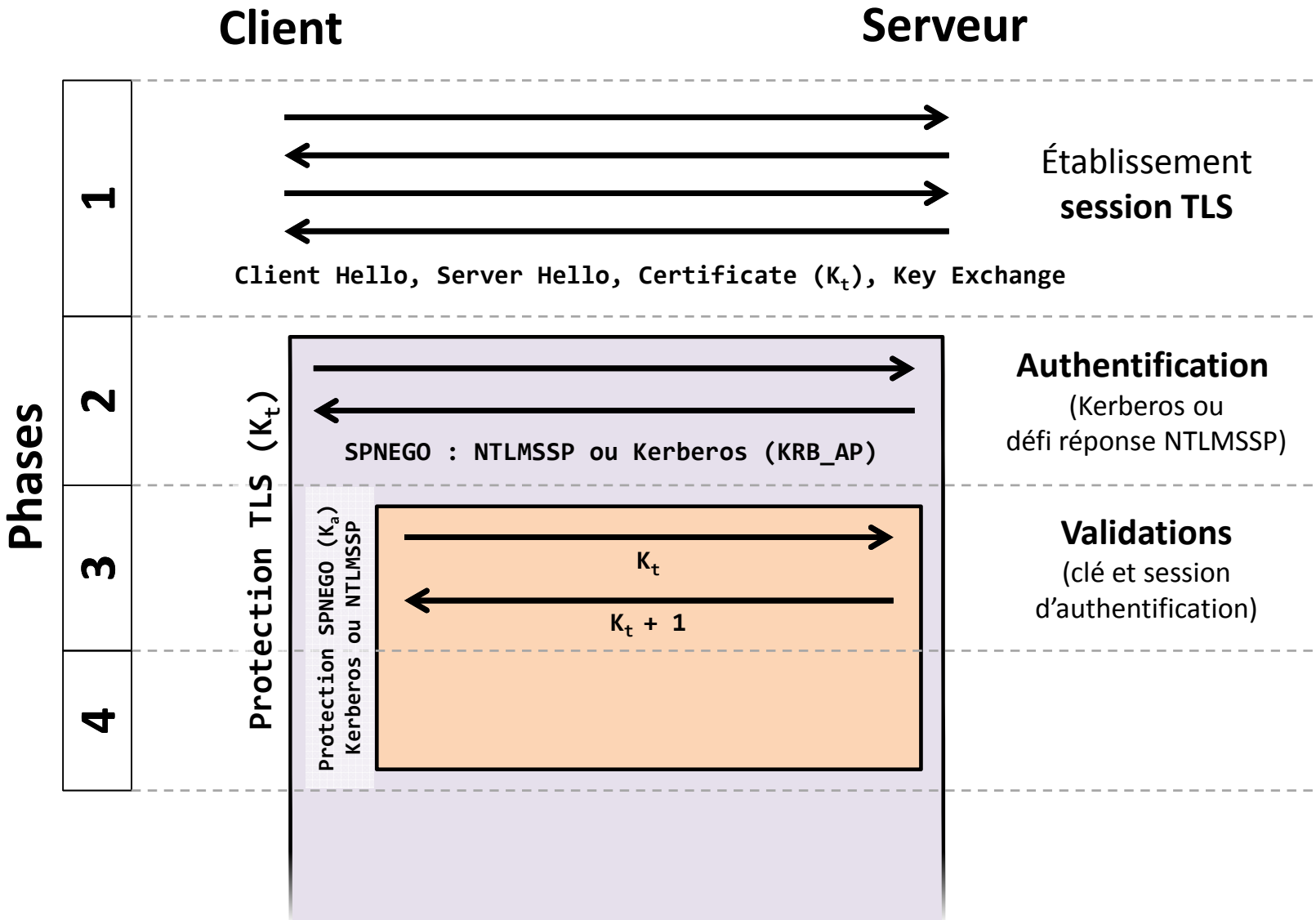
Couches protocolaires de NLA (1/4)



Couches protocolaires de NLA (2/4)



Couches protocolaires de NLA (3/4)



Couches protocolaires de NLA (4/4)

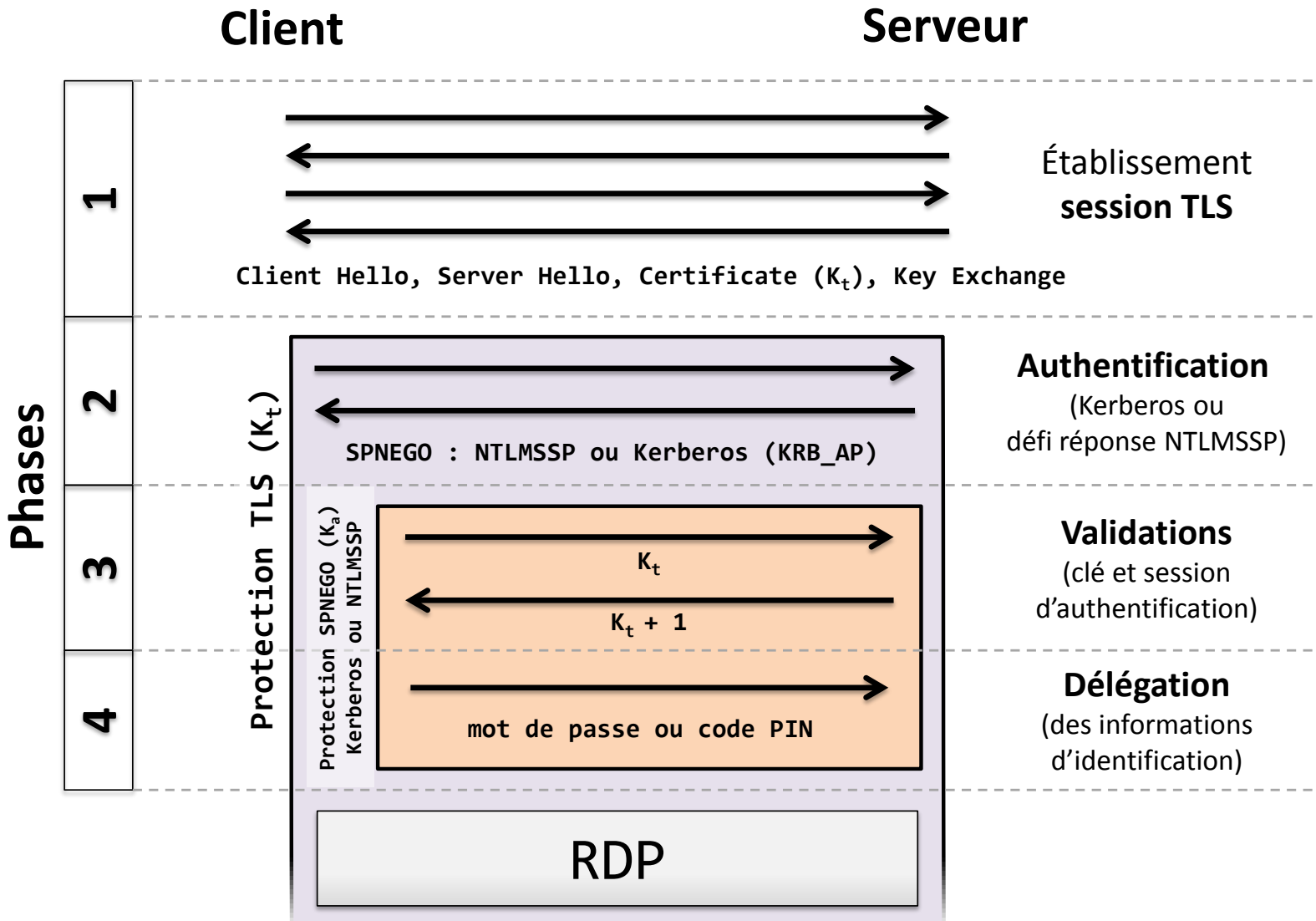
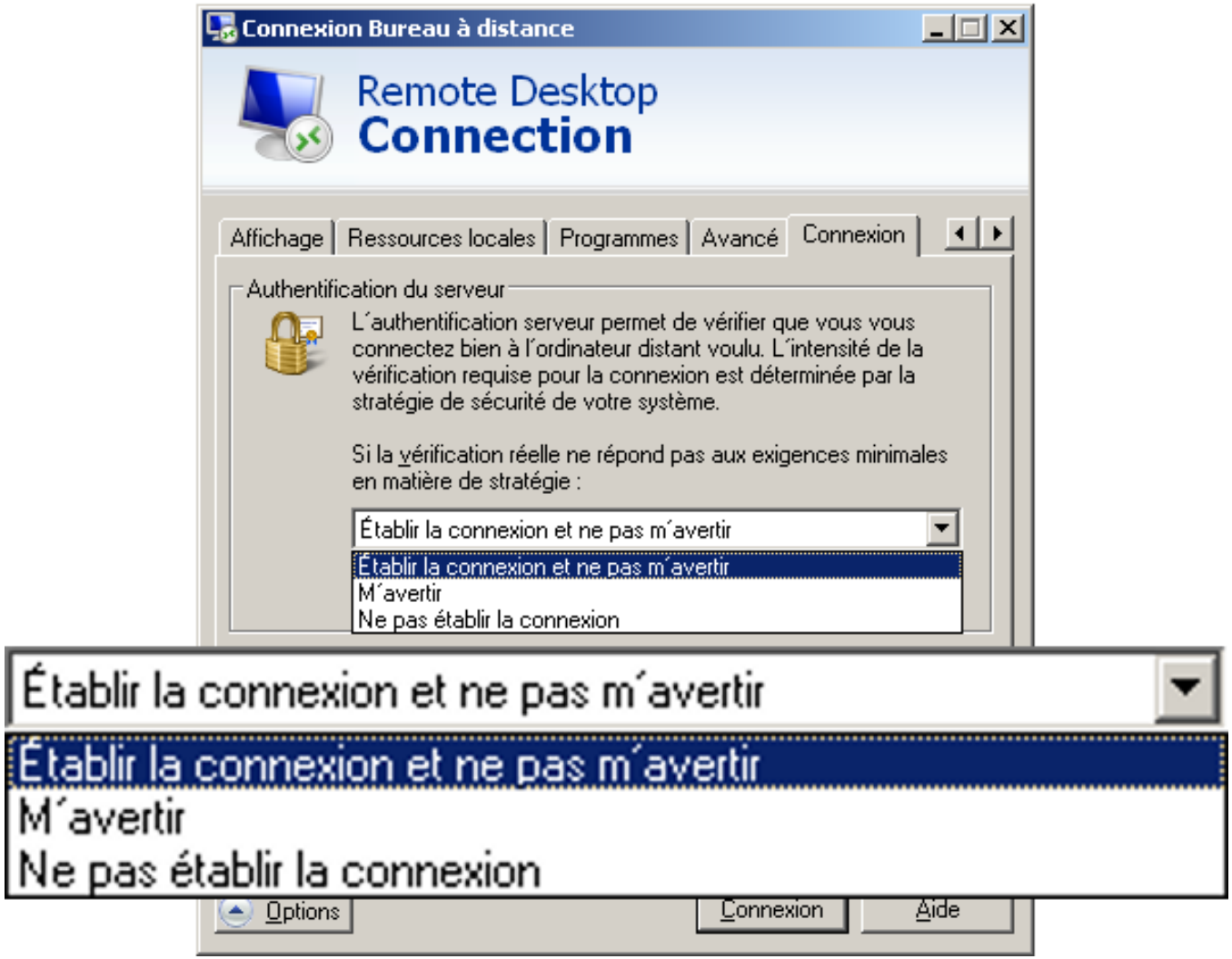
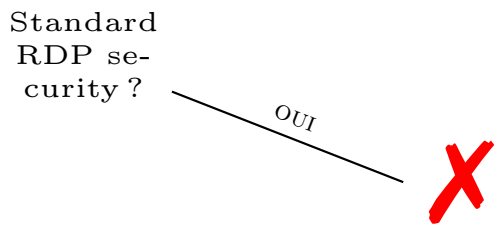


Table des matières

Options de configuration du client MSTSC

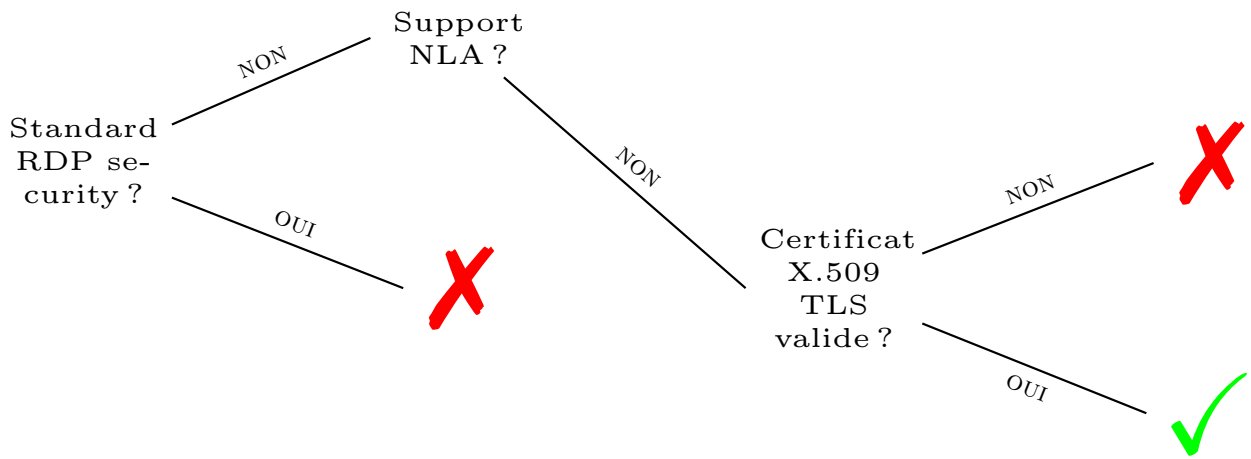


Logique d'authentification du serveur en *Enhanced*



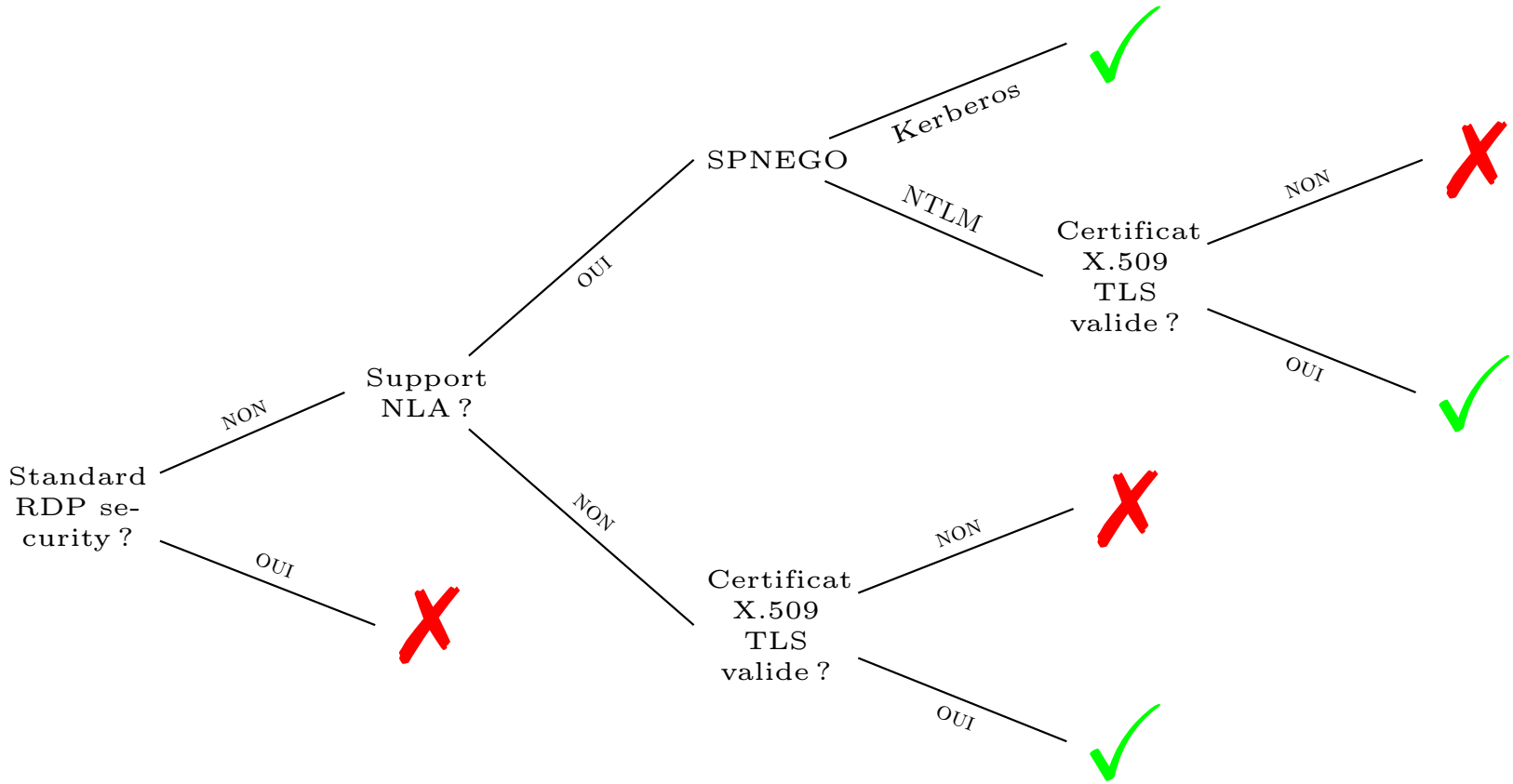
- ✓ : Serveur authentifié
- ✗ : Serveur NON authentifié
- Note : le client 6.0 considérait une connexion NLA en NTLM comme authentifiée

Logique d'authentification du serveur en *Enhanced*



- ✓ : Serveur authentifié
- ✗ : Serveur NON authentifié
- Note : le client 6.0 considérait une connexion NLA en NTLM comme authentifiée

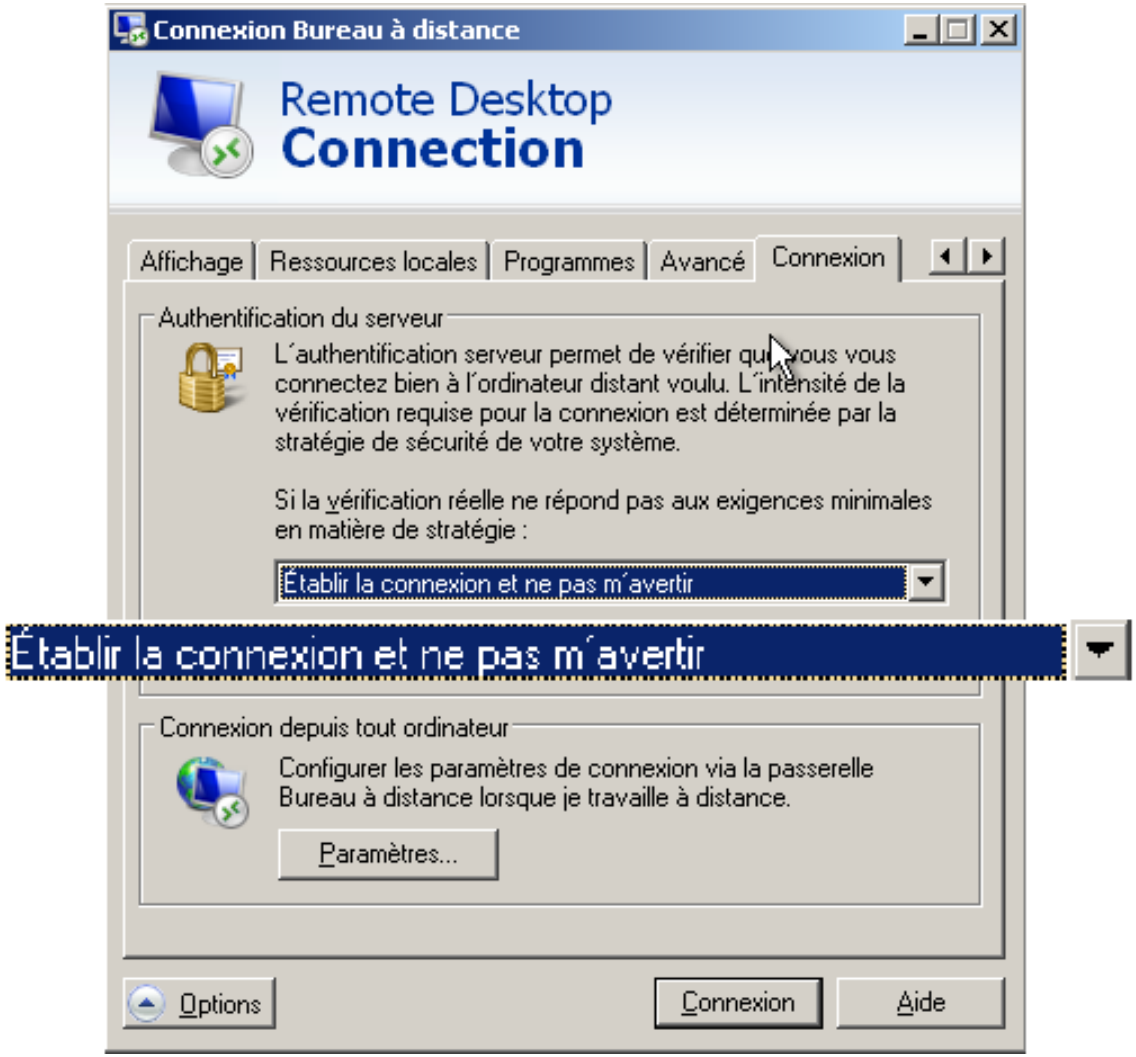
Logique d'authentification du serveur en *Enhanced*



- ✓ : Serveur authentifié
- ✗ : Serveur NON authentifié
- Note : le client 6.0 considérait une connexion NLA en NTLM comme authentifiée

Interlude récréatif #2

Configuration XP par défaut



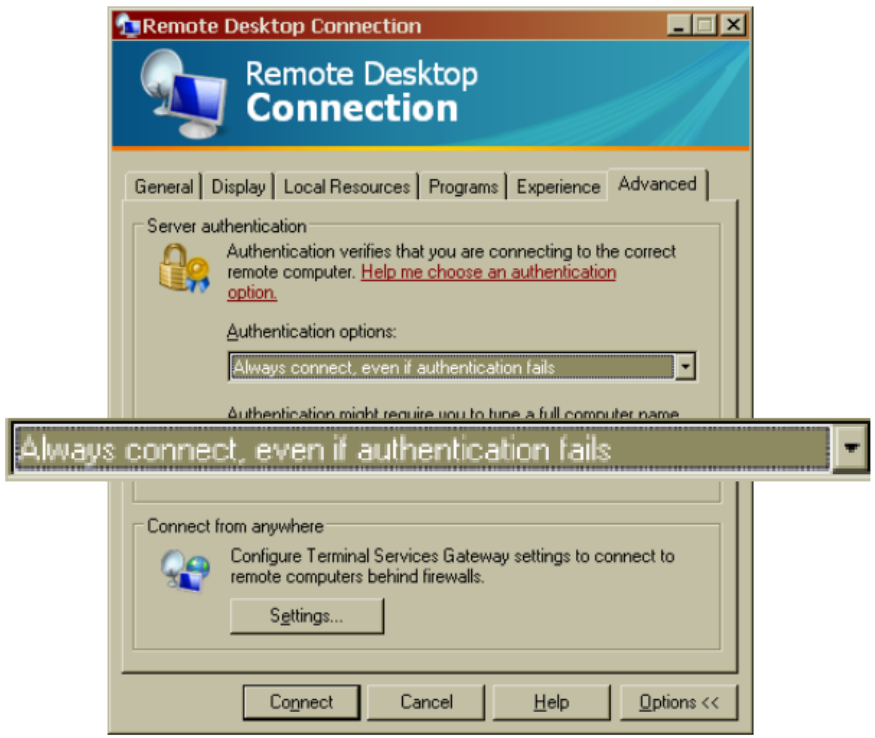
Interlude récréatif #3

Extrait du blog RDS [2]

How to eliminate the 'Remote Desktop cannot verify the identity of the computer you want to connect to..." messages

Answer: Before connecting, in Remote Desktop, do the following:

1. Click on "Options"
2. Click on the "Advanced Tab"
3. In "Authentication Options", select "Always connect, even if authentication fails, as seen below:



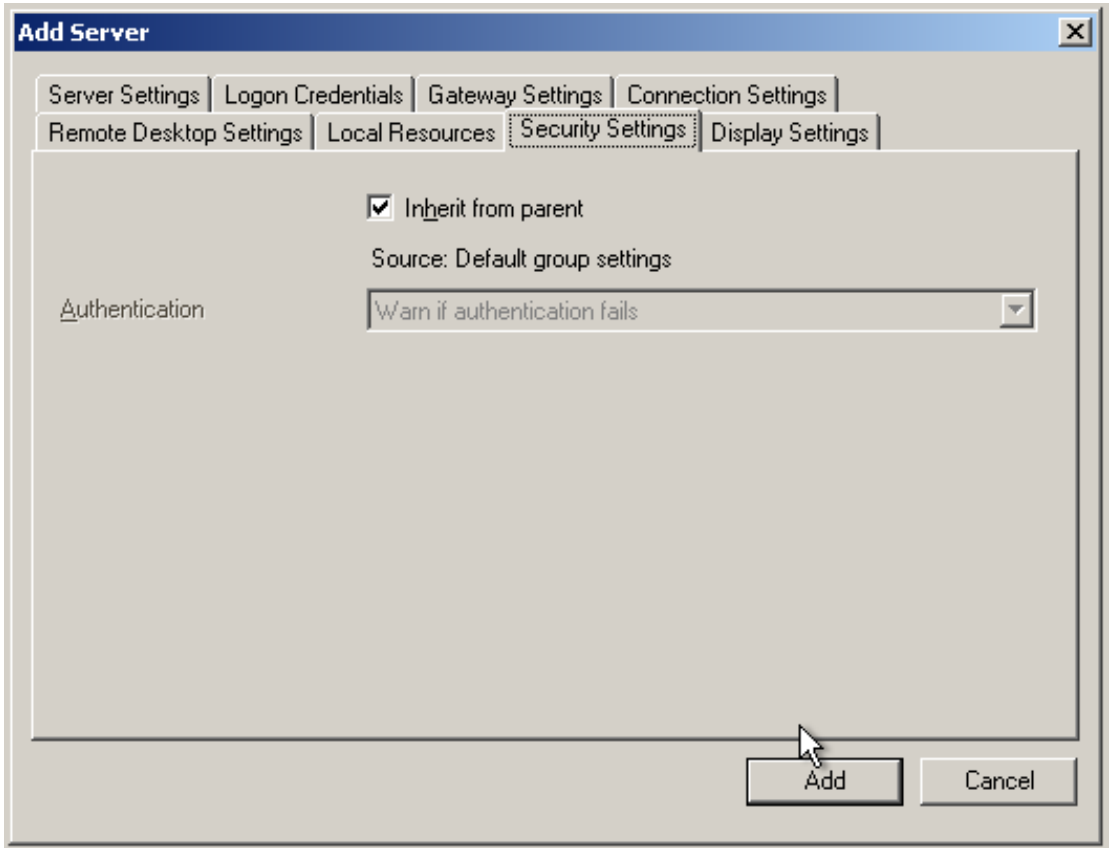
This will disable the warning prompt. Please be aware that selecting this option makes it possible for attackers to intercept and modify the data exchanged between client and server.

2. http://blogs.msdn.com/b/rds/archive/2007/01/22/vista-remote-desktop-connection-authentication-faq.aspx

Interlude récréatif #4

RDP connection manager

Vidéo présentant la logique d'héritage



Logique de connexion du client Microsoft

En NLA avec SPNEGO/NTLMSSP et l'option "M'avertir"

Logique de connexion du client Microsoft

En NLA avec SPNEGO/NTLMSSP et l'option "M'avertir"

1. connexion TLS **sans validation du certificat du serveur**
2. échange SPNEGO/NTLMSSP dans la session TLS
3. échange $K_t/K_t + 1$
4. validation du certificat contre les ancres du système

Logique de connexion du client Microsoft

En NLA avec SPNEGO/NTLMSSP et l'option "M'avertir"

1. connexion TLS **sans validation du certificat du serveur**
2. échange SPNEGO/NTLMSSP dans la session TLS
3. échange $K_t / K_t + 1$
4. validation du certificat contre les ancres du système
   - 4.1 ✓ certificat validé : session RDP authentifiée

Logique de connexion du client Microsoft

En NLA avec SPNEGO/NTLMSSP et l'option "M'avertir"

1. connexion TLS **sans validation du certificat du serveur**
2. échange SPNEGO/NTLMSSP dans la session TLS
3. échange $K_t / K_t + 1$
4. validation du certificat contre les ancres du système
   - 4.1 ✓ certificat validé : session RDP authentifiée
   - 4.2 ✗ certificat non validé : **déconnexion temporaire**

Logique de connexion du client Microsoft

En NLA avec SPNEGO/NTLMSSP et l'option "M'avertir"

1. connexion TLS **sans validation du certificat du serveur**
2. échange SPNEGO/NTLMSSP dans la session TLS
3. échange $K_t / K_t + 1$
4. validation du certificat contre les ancres du système
   - 4.1 ✓ certificat validé : session RDP authentifiée
   - 4.2 ✗ certificat non validé : **déconnexion temporaire**

5. dans ce dernier cas, tentatives additionnelles de validation :
   - alerte et acception du certificat par l'utilisateur **ou**
   - validation par rapport aux empreintes de la base de registre

Logique de connexion du client Microsoft

En NLA avec SPNEGO/NTLMSSP et l'option "M'avertir"

1. connexion TLS **sans validation du certificat du serveur**
2. échange SPNEGO/NTLMSSP dans la session TLS
3. échange $K_t / K_t + 1$
4. validation du certificat contre les ancres du système
   - 4.1 ✓ certificat validé : session RDP authentifiée
   - 4.2 ✗ certificat non validé : **déconnexion temporaire**

5. dans ce dernier cas, tentatives additionnelles de validation :
   - alerte et acception du certificat par l'utilisateur **ou**
   - validation par rapport aux empreintes de la base de registre

   en cas de succès :

6. **nouvelle** connexion TLS, **sans aucune authentification**

Logique de connexion du client Microsoft

En NLA avec SPNEGO/NTLMSSP et l'option "M'avertir"

1. connexion TLS **sans validation du certificat du serveur**
2. échange SPNEGO/NTLMSSP dans la session TLS
3. échange $K_t / K_t + 1$
4. validation du certificat contre les ancres du système
   - 4.1 ✓ certificat validé : session RDP authentifiée
   - 4.2 ✗ certificat non validé : **déconnexion temporaire**

5. dans ce dernier cas, tentatives additionnelles de validation :
   - alerte et acception du certificat par l'utilisateur **ou**
   - validation par rapport aux empreintes de la base de registre

   en cas de succès :

6. **nouvelle** connexion TLS, **sans aucune authentification**
7. échange SPNEGO/NTLMSSP
8. délégation des identifiants
9. établissement de la session RDP

Interlude récréatif #5

Vidéo

Recommandations (1/2)

Configuration

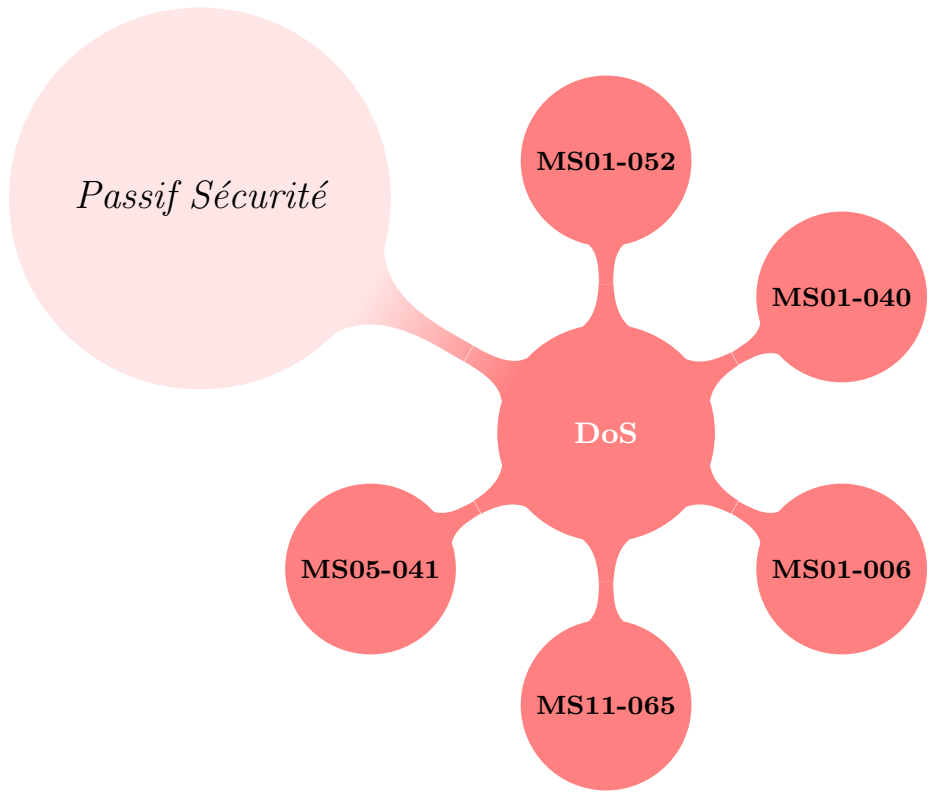
Seules options possibles côté client :

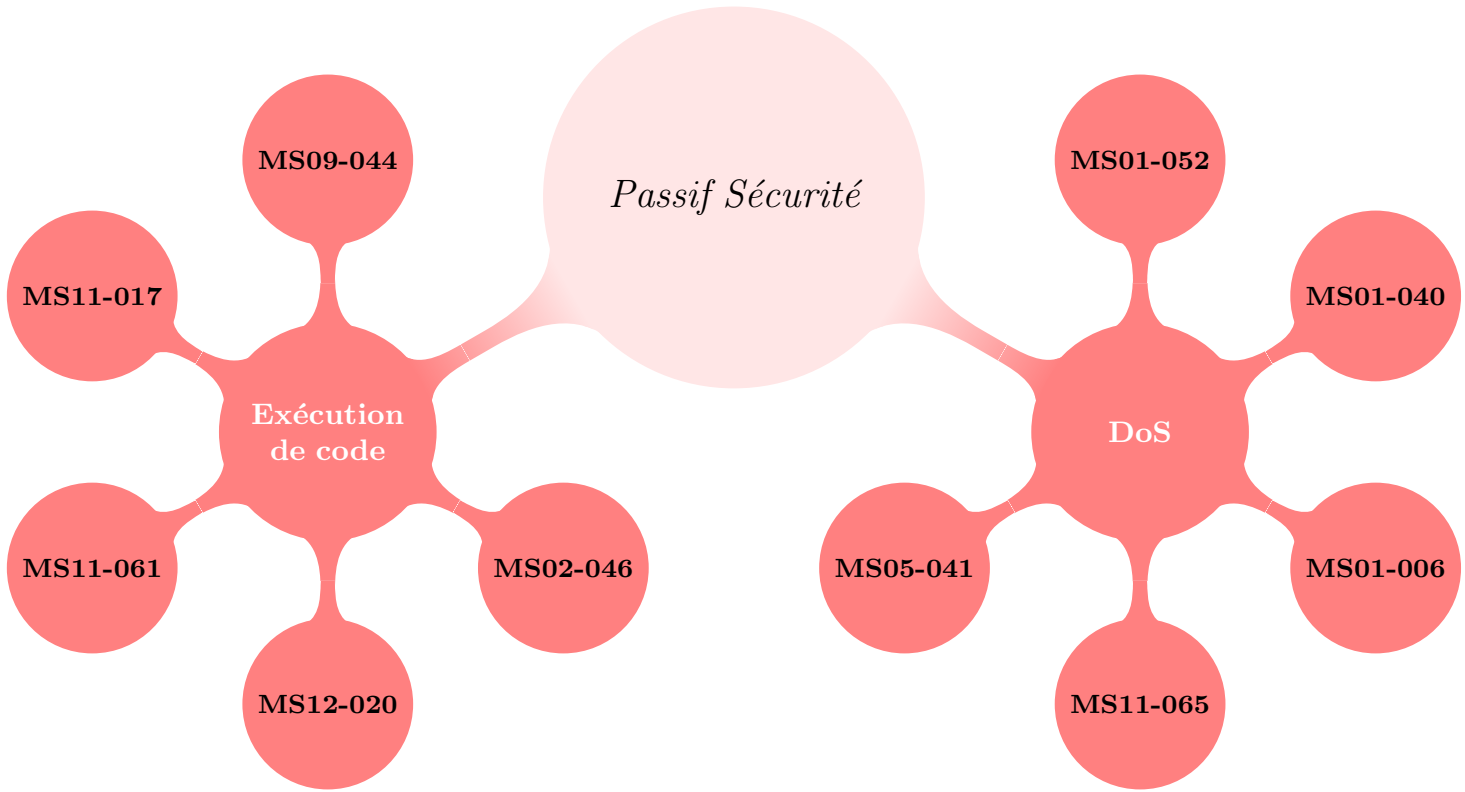
- installer la dernière version du client Microsoft ;
- **forcer l'authentification du serveur** (“*Ne pas établir la connexion*”).

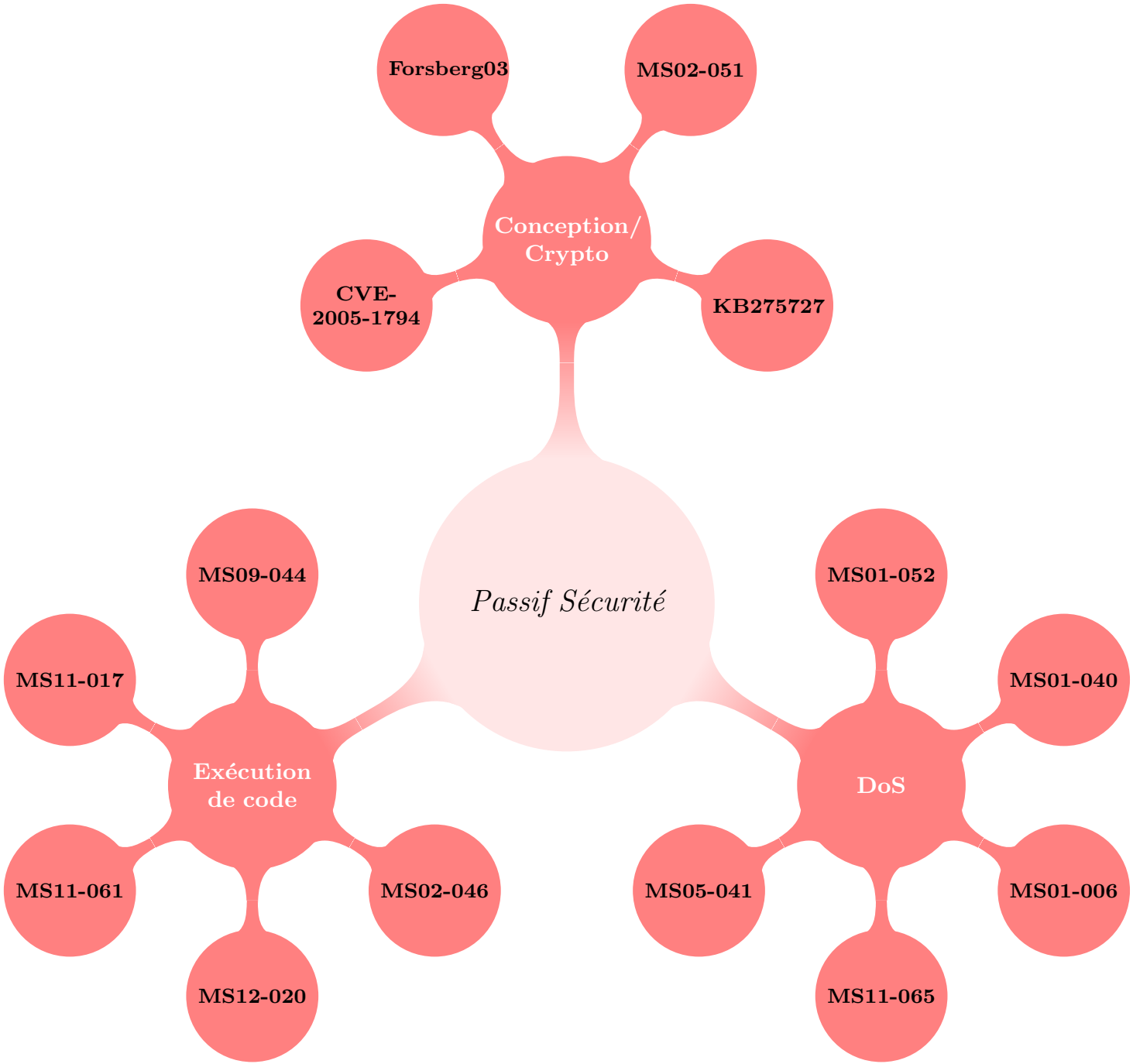
Côté serveur :

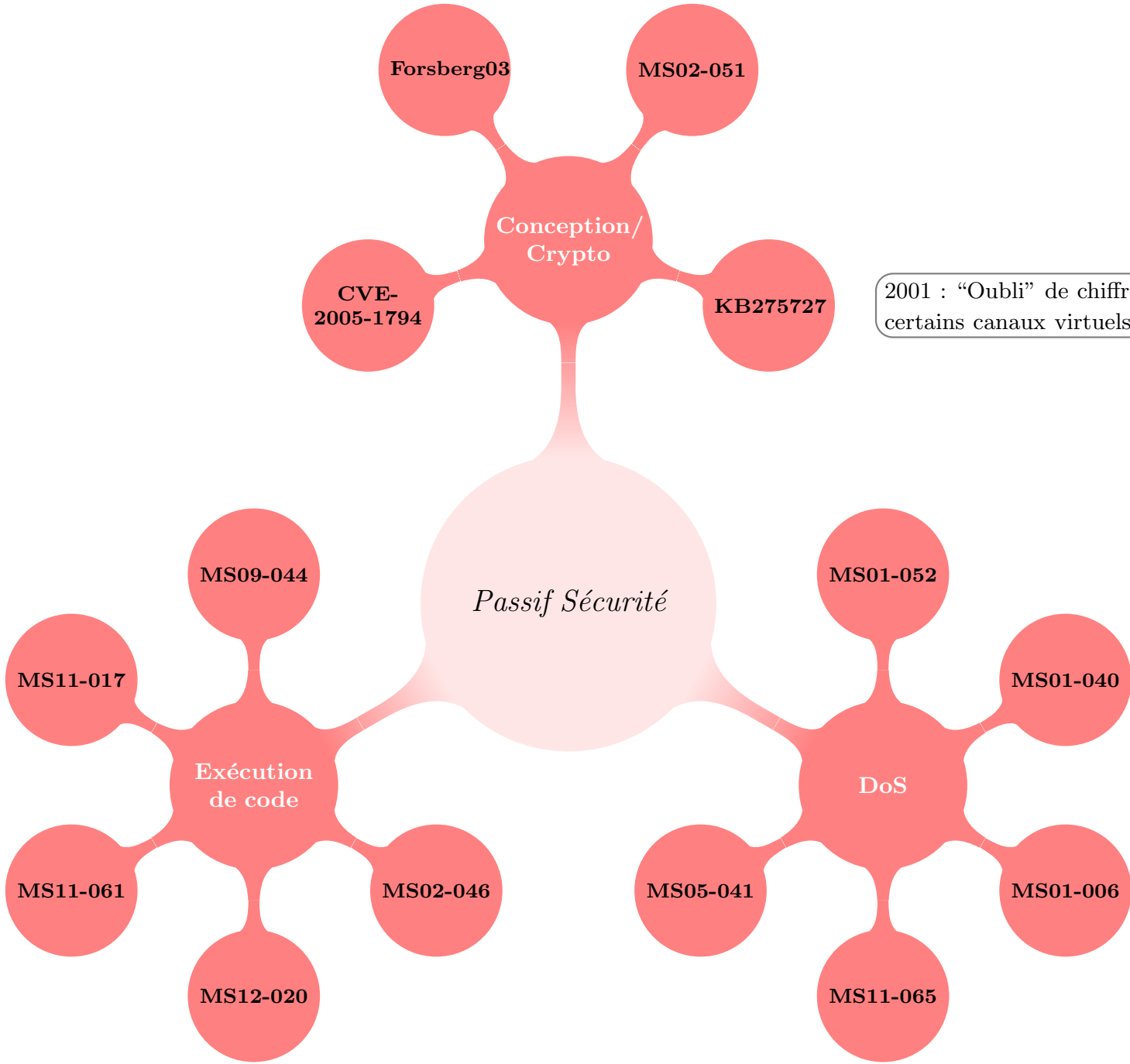
- mise à jour impossible ;
- en domaine :
  - XP : pas de salut ;
  - sur 2003 : activer et forcer TLS ;
  - sur Vista, Seven, 2008 : forcer NLA.
- hors domaine :
  - XP : pas de salut ;
  - autres : forcer TLS.

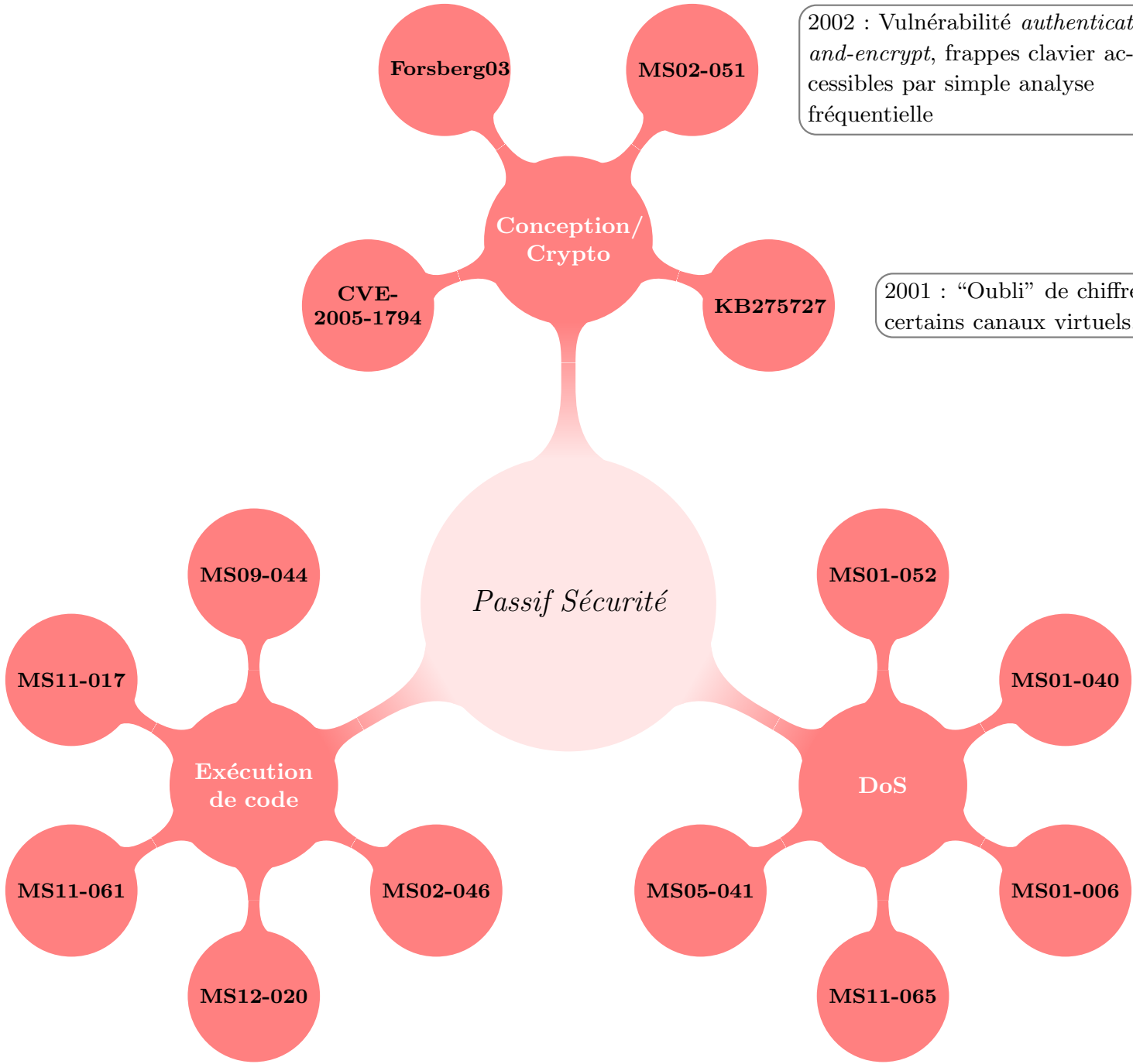
Passif Sécurité


Passif Sécurité
MS01-052
MS01-040
DoS
MS05-041
MS01-006
MS11-065


Passif Sécurité
Exécution de code
MS09-044
MS11-017
MS11-061
MS12-020
MS02-046
DoS
MS01-052
MS01-040
MS05-041
MS11-065
MS01-006

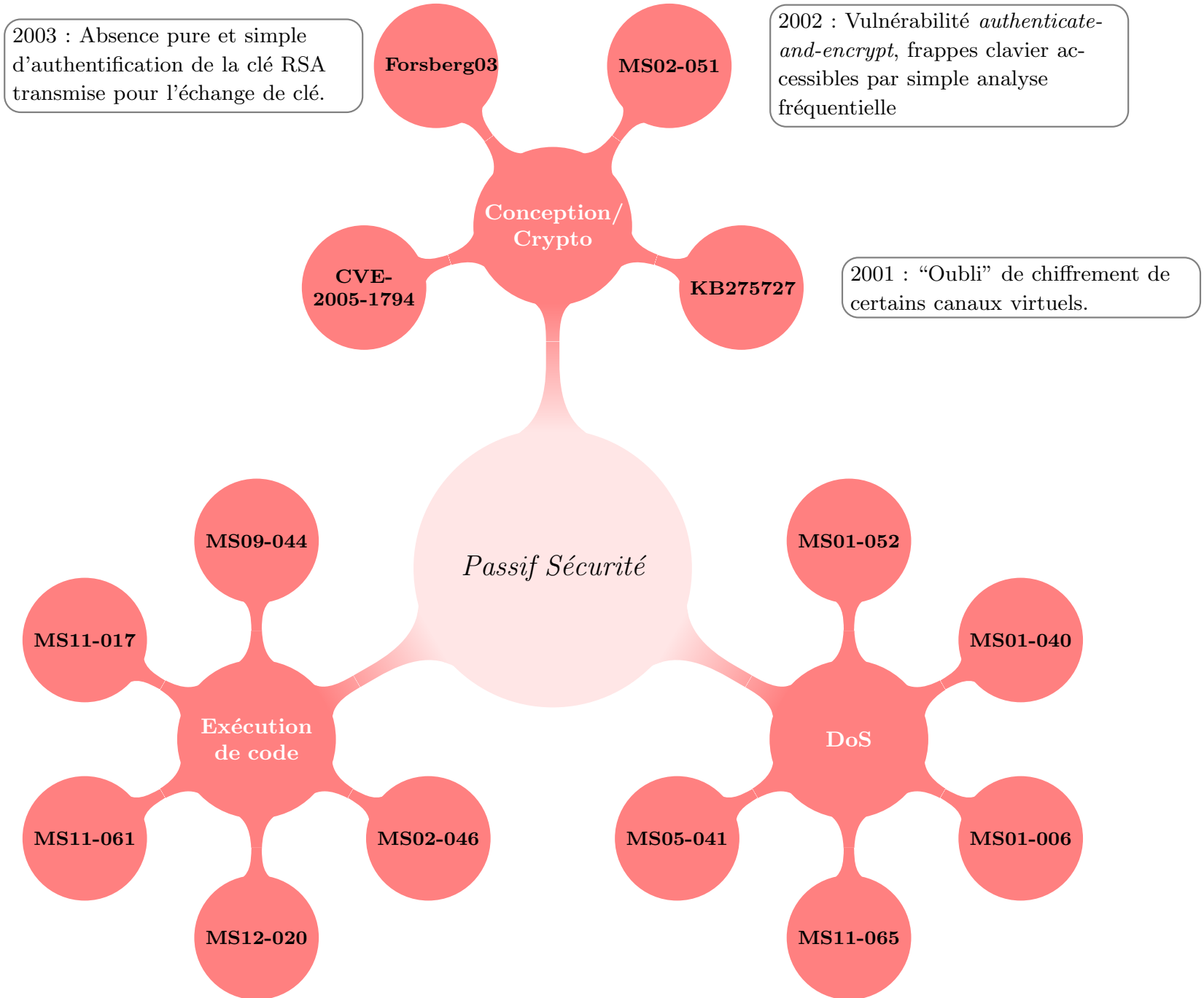

Forsberg03
MS02-051
Conception/
Crypto
CVE-
2005-1794
KB275727
MS09-044
MS11-017
Exécution
de code
MS11-061
MS02-046
MS12-020
Passif Sécurité
MS01-052
MS01-040
DoS
MS05-041
MS01-006
MS11-065

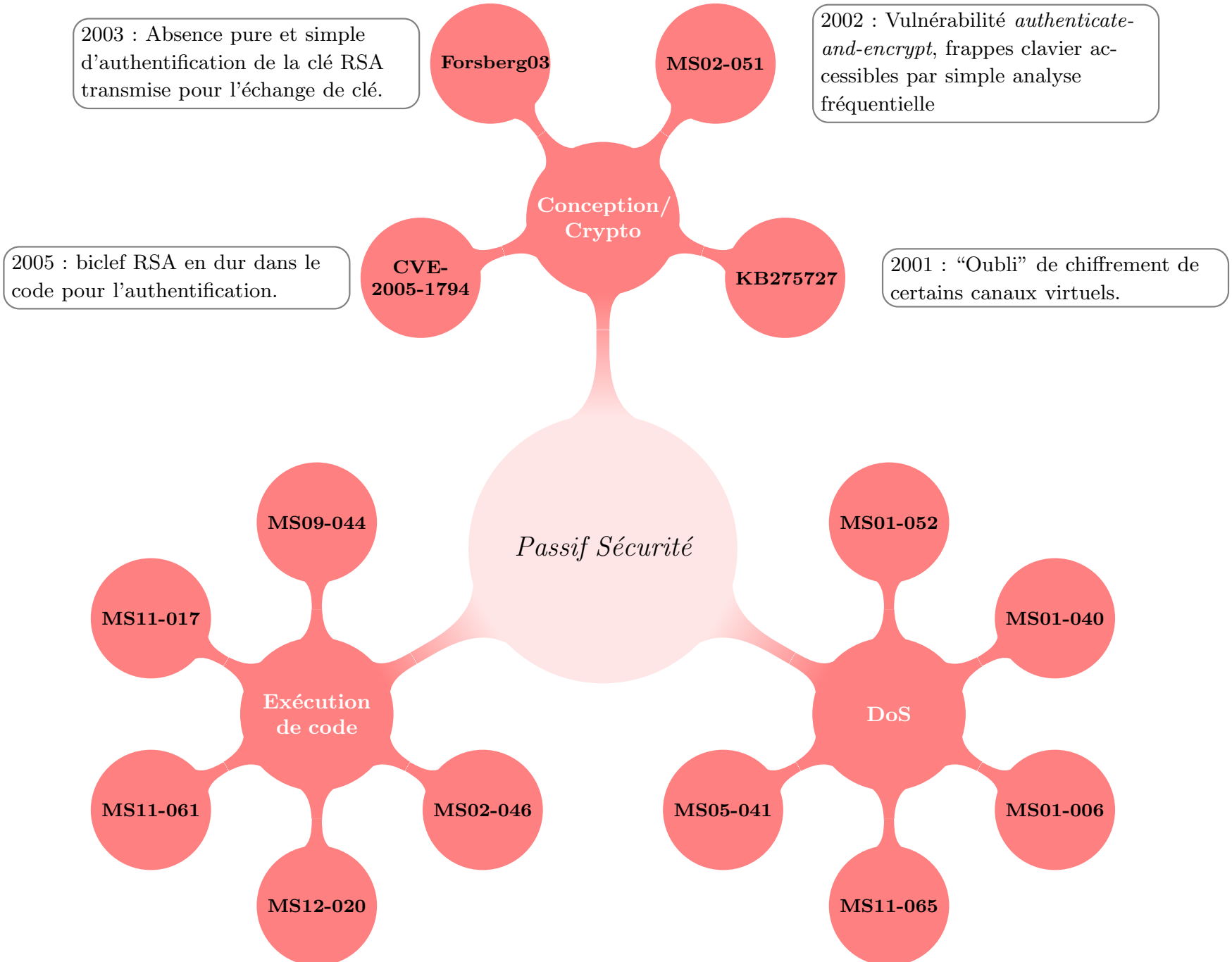

Forsberg03
MS02-051
Conception/
Crypto
CVE-
2005-1794
KB275727
2001 : “Oubli” de chiffrement de certains canaux virtuels.
MS09-044
MS11-017
Exécution
de code
MS11-061
MS02-046
MS12-020
Passif Sécurité
MS01-052
MS01-040
DoS
MS05-041
MS01-006
MS11-065



2002 : Vulnérabilité *authenticate-and-encrypt*, frappes clavier accessibles par simple analyse fréquentielle

2001 : "Oubli" de chiffrement de certains canaux virtuels.


2003 : Absence pure et simple d'authentification de la clé RSA transmise pour l'échange de clé.
Forsberg03
MS02-051
2002 : Vulnérabilité *authenticate-and-encrypt*, frappes clavier accessibles par simple analyse fréquentielle
Conception/ Crypto
CVE-2005-1794
KB275727
2001 : "Oubli" de chiffrement de certains canaux virtuels.
Passif Sécurité
MS09-044
MS11-017
Exécution de code
MS11-061
MS02-046
MS12-020
MS01-052
MS01-040
DoS
MS05-041
MS01-006
MS11-065


2003 : Absence pure et simple d'authentification de la clé RSA transmise pour l'échange de clé.
Forsberg03
MS02-051
2002 : Vulnérabilité *authenticate-and-encrypt*, frappes clavier accessibles par simple analyse fréquentielle
Conception/ Crypto
2005 : biclef RSA en dur dans le code pour l'authentification.
CVE-2005-1794
KB275727
2001 : "Oubli" de chiffrement de certains canaux virtuels.
Passif Sécurité
MS09-044
MS11-017
Exécution de code
MS11-061
MS02-046
MS12-020
MS01-052
MS01-040
DoS
MS05-041
MS01-006
MS11-065

Recommandations (2/2)

Architecture

Réseau :

- service RDP accessible uniquement aux administrateurs ;
- protéger les flux RDP d'attaques réseau.

Conclusion :

- Nécessité d'un réseau d'administration dédié.

Questions ?