

A thick dark blue vertical bar runs down the left side of the slide. A blue arrow-shaped banner points to the right from this bar, containing the date. In the bottom left corner, several thin, curved lines in dark blue and light grey sweep upwards and to the right.

3/21/2016

Patterns de sécurité

Cas pratique : modélisation de patterns
de sécurité et de l'architecture abstraite
SCADA

Zyad ALAMRI
Muhannad ALHUWAIMIL
ENSTA BRETAGNE

Table of Contents

Table des figures	2
Introduction	3
A. Diagramme de cas d'utilisation du Scada sécurisé :	4
I. Diagramme de séquences du pattern « Autorisation »	5
II. Diagramme de séquences du pattern « Single Access Point »	5
III. Diagramme de séquences du pattern « Check Point »	5
C. Diagrammes de classes	7
D. Machines à états	8
E. Diagrammes d'Objets	9
F. Difficultés Rencontrés	10
I. Installation	10
II. Connexion	10
I. Mini-Conclusion	11
Conclusion	13

Table des figures

Figure 2: Problème de Validation de la license de Rhapsody	10
Figure 3: Problème de connexion vpn – 23h28.....	11
Figure 4:/Problème de connexion vpn – 07h02	11

Introduction

La sécurité a toujours été une question prioritaire pour les gouvernements, les opérateurs financiers et les entreprises. Elle fait de plus en plus partie de notre quotidien surtout au regard de la mondialisation et des objets connectés.

Le piratage que ce soit via des accès physiques, des interceptions de connections, des dénis de services ou alors des intrusions fait partie du quotidien et de l'interrogation perpétuel des spécialistes du métier.

La mondialisation et la connectivité accrue sur tous les plans : objets connectés, smarthome, smartcity obligent à revoir les techniques de sécurisation. C'est l'une des raisons pour laquelle on a eu l'opportunité d'acquérir des connaissances théoriques sur les patterns de sécurités.

Afin de montrer à quel point on a compris mais aussi apprécié ce magnifique cours, il nous a été demandé ainsi qu'aux camarades de classes de mettre en œuvre des patterns de sécurité.

A cet effet, on a modélisé l'architecture abstraite du système SCADA. Puis une modélisation de la dite architecture composé avec 3 patterns (Authorization, Single Access, Check Point).

Dans tous ce qui suit, on considèrerait un Scada sécurité un système scada composé des trois patterns énoncés précédemment.

Afin de réaliser le modèle exécutable, je vais présenter les 5 cinq catégories de diagrammes réalisés. Il s'agira donc du diagramme de cas d'utilisation, de séquences et de classes. Des machines à états de différents composants et d'un diagramme d'objets permettant de générer les liens et de connexion pour l'exécution du modèle.

A. Diagramme de cas d'utilisation du Scada sécurisé :

Cette partie est consacrée au diagramme du cas d'utilisation de Scada sécurité. Etant donné qu'on se focalise sur la sécurité, je ne peux me soustraire qu'au simple système Scada. On se focalisera donc sur cet aspect sécurité.

Dans l'incapacité d'utiliser Rhapsody, je ne pourrais malheureusement pas décrire cette partie. On vous prie de vous référer exclusivement au diagramme UML. On avait rajouté des notes explicatives. Merci pour votre compréhension.

Pour des raisons de connexion et dans la suite du rapport, on vous prie de vous référer aux diagrammes directement dans Rhapsody.

B. Diagramme de séquences

Trois diagrammes de séquences feront parties intégrales de cette partie. Bien sûr, on a omis principalement le diagramme de séquence du Scada et du Scada sécurisé car le but principal est l'aspect sécurisé. En effet, il n'y a pas lieu de se visualiser les scénarii d'exécution spécifique au système Scada. Ce qui importe sont ceux faisant intervenir les patterns cités en Introduction.

I. Diagramme de séquences du pattern « Autorisation »

Le pattern **Autorisation** est un pattern qui limite les accès (ressource) selon certains critères du système. De plus il structure les différentes possibilités d'accès.

Ce diagramme décrit spécifiquement deux scénarii possibles:

- La première pour lui permettant d'accéder aux ressources systèmes et de communiquer avec le dit système. Les accès pouvant être de plusieurs types : *read*, *write* ; avec contraintes.
- La seconde pour laquelle une permission lui ai refusée.

En effet, lorsqu'un sujet veut accéder au système, il déclenche un évènement *requestAccess()* afin de se rassurer que le sujet a le droit d'accéder au dit système. Une fois l'évènement reçu par la classe **CheckPoint**, un évènement de vérification des droits est déclenché (*checkRight()*). Deux options sont alors possibles :

- Si vérification s'avère positive (évènement *approved()* reçu), la connexion est établie et le sujet peut dès lors interagir avec le système.
- Dans le cas où l'évènement *rejected()* est reçu, le sujet est notifié.

II. Diagramme de séquences du pattern « Single Access Point »

Ce pattern limite les accès à un seul point d'entrée tout en capturant les informations de log afin de garder une trace. C'est la raison pour laquelle une seule séquence d'exécution est nécessaire. Celle pour laquelle un client émet une requête d'accès en déclenchant une transition *access()* vers la classe **SingleAccessPoint** qui à son tour place effectue un enregistrement dans un dossier client.

III. Diagramme de séquences du pattern « Check Point »

Il est différent du pattern **Autorisation** dans ce sens que lui contrôle le type d'accès selon une politique de sécurité et des prend mesures si besoin (en cas de violation). D'ailleurs, le pattern **Autorisation** fait appel au pattern **CheckPoint** via la classe **CheckPoint** comme présenté

dans la séquence d'exécution de du pattern **Autorisation** de la section **Diagramme de séquences du pattern « Autorisation »**.

Vous l'aurez deviné, l'idée reste la même mais les séquences d'exécution sont légèrement différentes. Les scénarii présentés ici sont ceux offrant des accès où pas.

Il est temps de vous présenter de vous les présenter. Supposons qu'un composant externe veuille communiquer avec le système. Il déclenchera de facto l'évènement message().

C. Diagrammes de classes

A chaque brin du système, correspond un diagramme de classes. Ainsi j'ai modélisé cinq diagrammes de classes. Trois correspondants aux patterns de sécurité et deux correspondants au système Scada seul et Scada sécurisé.

D'autre part on ne décrirait pas les diagrammes en questions tous simplement parce que cela reviendra à redire ce qui a été dit graphique via les liens de d'associations.

D. Machines à états

La modélisation du système ne peut passer par la description du comportement des composants du dit système. Ceci dit, à chaque classe, correspond un comportement. Ceci vaut aussi pour les acteurs du système.

C'est l'une des parties les plus cruciales mais elle est impossible à décrire à la volé du moins sans avoir les captures d'écran ? On s'en excuse profondément.

E. Diagrammes d'Objets

Le diagramme d'objet est un diagramme de classes où les éléments liés sont les objets et donc les instances des classes. Ceci permet de générer les liens entre objets afin d'exécuter le modèle décrit.

F. Difficultés Rencontrés

Un projet sans difficultés est de l'ordre de l'utopie. Comme certains professeurs le disent tantôt, l'important est de pouvoir les surmonter afin d'obtenir un résultat. Ainsi, on a jugé judicieux de présenter les difficultés rencontrées surtout au regard du fait que ça a été un point bloquant lors de la réalisation de ce projet.

I. Installation

Notre calvaire a commencé avec l'installation de Rhapsody. L'installation en elle-même n'a pas été problématique sinon on se poserait la question de savoir comment on a bien fait pour atterrir dans une école d'ingénieur. Le problème en question était l'exécution du logiciel qui comme tous logiciels nécessitent une licence. On pensait que la clé était intégrée avec le logiciel.

Un premier contact a eu lieu avec l'enseignant et on a ainsi été en possession d'une licence serveur. Mais cela n'a pas résolu mon problème comme le confère la figure ci-dessous.

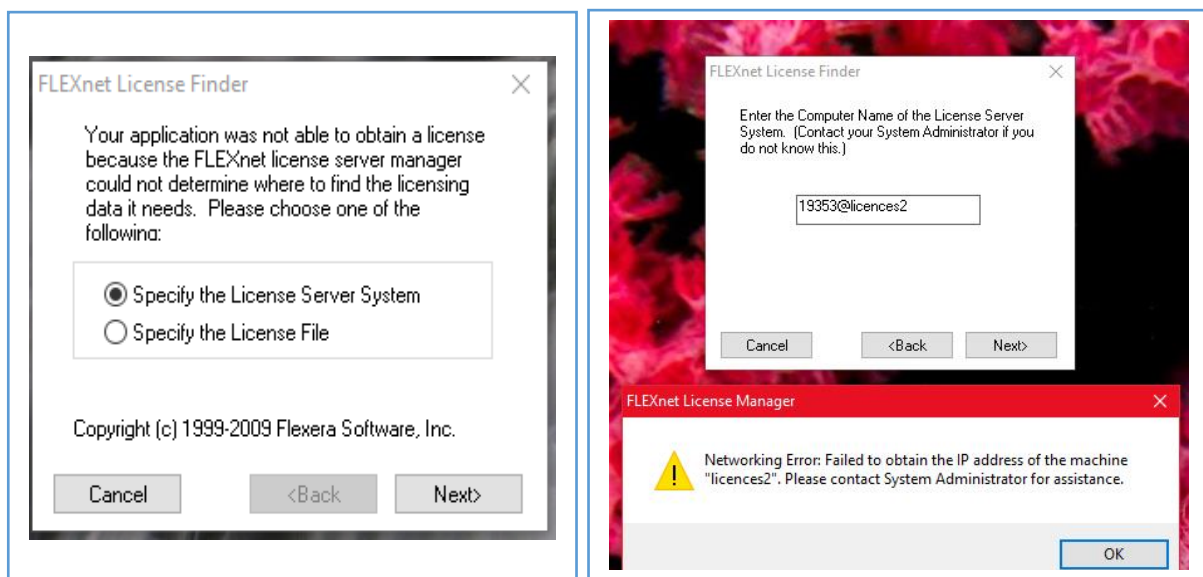


Figure 1: Problème de Validation de la licence de Rhapsody

II. Connexion

Une sous difficulté est apparue et c'était celle de l'utilisation de la licence serveur. En effet, la licence n'était pas censée fonctionner sur un pc connecté à internet. Un retour vers l'enseignant m'a permis de comprendre qu'il fallait être sur le réseau local de l'ENSTA Bretagne d'où la mise en place d'un VPN qui nous permettrait de me connecter aux serveurs de l'école.

Ceci dit, on s'est heurté au fur et à mesure de mon développement à une impossibilité de connexion en vpn et ceux d'après mon décompte pendant près de deux semaines plein que

ce soit en matinée (6h – 7h) comme en soirée (19h-00h). Ci-dessous, deux captures d'écran liées à mon impossibilité d'utiliser Rhapsody : une à 23h28 et une autre à XXX.

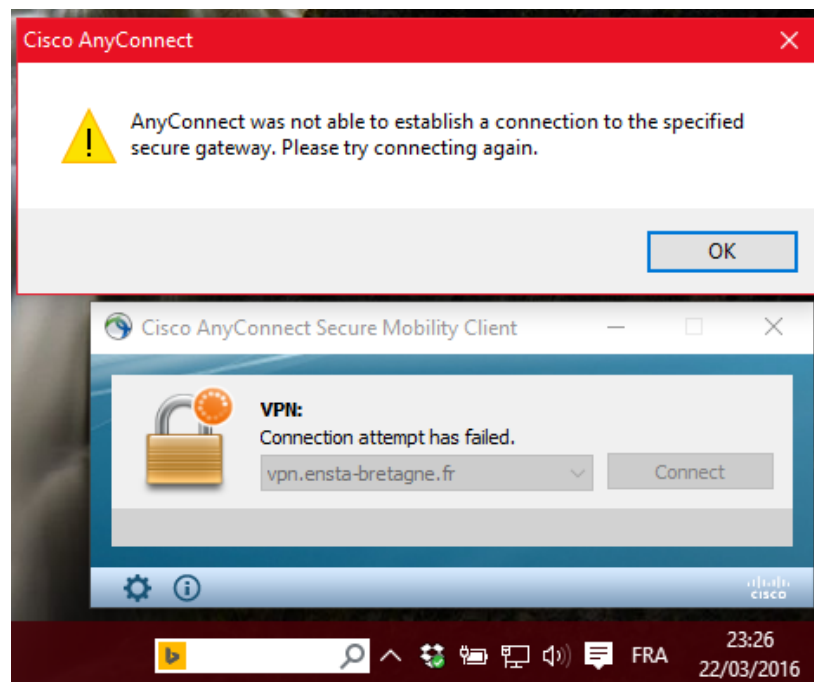


Figure 2: Problème de connexion vpn – 23h28

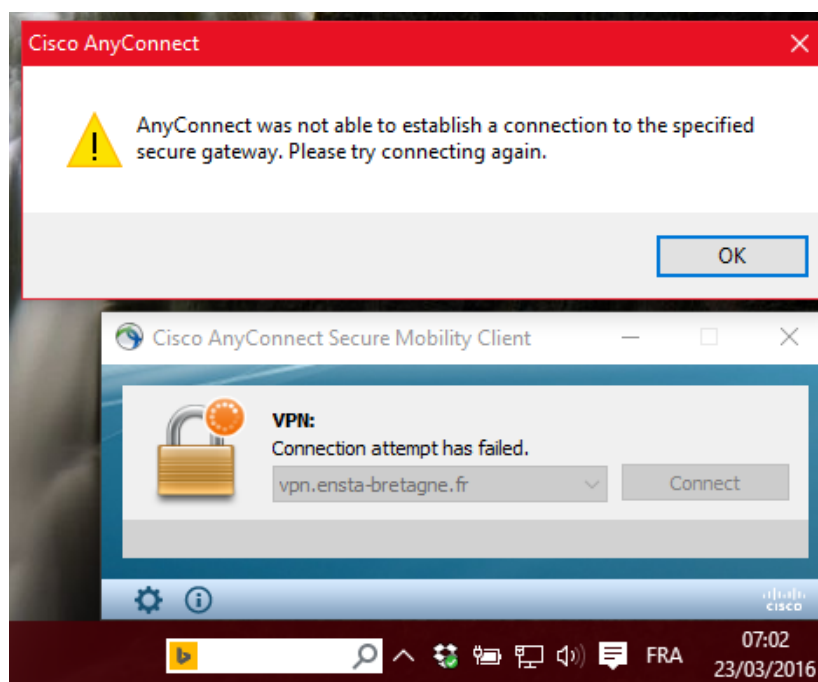


Figure 3: Problème de connexion vpn – 07h02

I. Mini-Conclusion

Les difficultés liées aux projets informatiques sont toujours existantes. Elles peuvent être de toutes sortes : installation, configuration, utilisation, codage et autres.

Les plus frustrantes sont celles pour lesquelles nous n'avons pas beaucoup d'options. En l'occurrence l'installation, et sa disponibilité. Ce sont des risques qui s'avèrent bloquantes.

Mes propositions pérennes sur ce sujet sont les suivantes : l'utilisation d'une machine virtuelle contenant Rhapsody.

Conclusion

Cette unité d'enseignement m'a permis de comprendre non seulement l'importance des patterns, leurs processus mais aussi comment ils sont associés ou peuvent être associés aux systèmes en particulier le système Scada.

On a adoré travailler sur ce projet de pattern de sécurité et on aurait aimé consacrer encore plus de temps sur ce type de matières surtout au vue de la digitalisation croissante.

Sans toutefois vous vexer, on aimerait aussi qu'à l'avenir on consacre un peu plus de temps sur le système Scada car on a eu un tel intérêt et par ricochais des interrogations pour lesquelles google n'a pu y répondre.

Une fois de plus merci,
ZyadALAMRI,
Muhannad ALHUWAIMIL.