Uma arquitetura para desenvolvimento de dispositivos de autenticação e acesso a espaços físicos

Jeann C. M. Raguzzoni, Lamarck Ribas Heinsch
UFSM
jraguzzoni@inf.ufsm.br,lamarck@inf.ufsm.br

Tiago Antônio Rizzetti
UFSM - CTISM
rizzetti@ctism.ufsm.br

***Resumo*—Ambientes de trabalho e estudo estão cada vez melhor equipados e sofisticados, sendo necessário um rastreamento temporal sobre acessos realizados a estes ambientes.. Esta arquitetura se propõe a resolver este problema através da implementação de um controle de acesso eletrônico com diretivas e bases de acesso controladas por um sistema de alto nível, baseado em redes TCP/IP, que facilite aos administradores do local gerenciar o fluxo de pessoas autorizadas.**

I. Introdução

Grandes instituições geram um grande fluxo de pessoas, um fluxo difícil de ser controlado. Na tentativa de conter acessos utiliza-se bloqueios a certas áreas. Em uma faculdade, por exemplo, temos incontáveis salas de aula, laboratórios, salas de professores, etc. Estas salas são, na maioria das vezes, mantidas fechadas por chave, a qual é retirada na portaria, ou outro setor especializado, e depois devolvida, muitas vezes sem passar por uma verificação do usuário ou registro de uso. Este é um meio de controle precário, ineficiente e inseguro.

Como uma proposta para solução deste problema, criou-se a arquitetura do projeto ESC (Environment Security Control). Este, consiste em uma arquitetura de controle de acesso diferenciada, focada em uma abordagem que possibilite a comunicação entre os dispositivos de autenticação, junto ao acesso do espaço físico, e uma aplicação gerente, podendo desta forma ampliar a capacidade de configuração e coerência do sistema. Em função desta abordagem, aumenta-se de forma expressiva os aspectos referentes a escalabilidade, flexibilidade e auditoria. Dividiu-se este sistema sobre dois aspectos principais, um considerando o gerente, o qual é responsável pelo tipo de politica empregada para autenticação e todo o ambiente de integração com os demais sistemas utilizados, como o LDAP; o segundo subsistema, no qual se encontra o foco deste trabalho, projeta e implementa uma interface de hardware e software capaz de lidar com os dispositivos físicos de autenticação. Obtendo seus dados e, através de um protocolo criado, comunicando-se com o gerente de forma bidirecional. Desta forma possibilitando além de sensoriamento do ambiente, a capacidade de realizar ações sobre o sistema através de atuadores.

O restante deste trabalho terá suas informações focadas no método e protocolo de comunicação entre o dispositivo de hardware remoto e o sistema gerente de alto nível, os dois subsistemas da arquitetura apresentada a seguir.

II. Trabalhos e Sistemas Relacionados

Os trabalhos relacionados foram separados em duas categorias, soluções proprietárias e soluções acadêmicas. As soluções proprietárias consistem em produtos desenvolvidos por empresas de médio ou grande porte, geralmente especializadas na área de segurança. Essas empresas investem pesado em pesquisa e desenvolvimento de modo a oferecer soluções robustas. A vantagem de utilizar uma solução dessas é o fato de haver maior compatibilidade entre equipamento provido pela empresa e o sistema gerenciador de permissões de acesso. Porém existem desvantagens, como por exemplo, o produto ser vendido em kits, isto é, preços definidos de acordo com a quantidade de dispositivos tornando o custo de aquisição e manutenção alto. Outro problema comum é o código-fonte ser fechado, impossibilitando a correção de erros no código, implementação de novas funcionalidades e adaptação a necessidades do cliente

Dentre as soluções proprietárias, foram analisadas duas empresas, Nibtec [10] e ID Tech [9]. Estas empresas diferem em tipos de dispositivos de acordo com o método de entrada (login/senha, biometria ou smartcards) e na estrutura utilizada entre o software gerente dos dispositivos físicos. Entretanto, por serem empresas que a grosso modo, têm em seu escopo de negócios, clientes de grande porte, ocultam ao máximo seus códigos-fonte afim de dificultar a busca por falhas eventualmente exploradas em uma entrada furtiva e também para manter a propriedade intelectual devido à grande quantidade de pesquisas para gerar tal produto. Além do custo pago pelos dispositivos físicos de acesso, algumas dessas empresas cobram um valor extra, no caso do cliente desejar possuir o software gerente de permissões e restrições, pois é possível possuir apenas o modo de autenticação no próprio dispositivo.

No que concerne as soluções acadêmicas, foram pesquisados dois trabalhos de origem acadêmica, o projeto Sentinel e um trabalho relativo a controle de acesso ao prontuário eletrônico de pacientes.

O projeto Sentinel [8] é um sistema em Java para controle de acesso baseado em papéis. Existe um módulo de auditoria que registra os acontecimentos, porém a ênfase do trabalho é a autenticação. O Sentinel é baseado no conceito de plug-ins, onde é o método de autenticação é desenvolvido de acordo com a necessidade de cada dispositivo físico, tornando-o assim, versátil. Entretanto, é apresentado apenas trechos de códigos-fonte e diagramas UML, não sendo encontrado o código-fonte do sistema.

Já o outro projeto [11], visa permitir acesso aos prontuários de pacientes evitando grandes burocracias e de modo fácil, contrário ao que existe atualmente, que é baseado em papéis. Para isto, foi desenvolvido um sistema em Java que faz acessos a uma base LDAP. Este sistema tem a vantagem de ser um software livre, baseado em componentes sem custos de licenciamento e visa

apresentar bom desempenho para as demandas de acesso do mesmo.

Tendo em vista a dificuldade de flexibilidade, adaptação, introdução de novas categorias de dispositivos físicos e dependência de fornecedores/fabricantes específicos, incluindo altos custos que as soluções proprietárias apresentam, é proposto neste trabalho a arquitetura ESC, buscando construir uma solução utilizando apenas ferramentas livres.

III. A arquitetura ESC

A arquitetura ESC (Enviroment Security Control) foi desenvolvida com o intuito de facilitar o gerenciamento de acesso e controle de ambientes restritos. Seu destaque principal é a facilidade de comunicação entre diversos dispositivos de interação com o meio físico e um gerente centralizado de alto nível. Isto é feito através de uma abstração do tipo de dado de entrada, criando, desta maneira, uma grande transparência entre o meio físico e a administração do sistema. A arquitetura foi dividida em dois subsistemas: ESCHA (ESC Hardware), qualquer dispositivo remoto, e ESCMA (ESC Manager), o gerente centralizado para processamento das informações. A conexão entre os dois ocorre através de interfaces de rede ethernet, usando a pilha de protocolos TCP/IP. Uma ilustração desse sistema pode ser vista na figura 1.

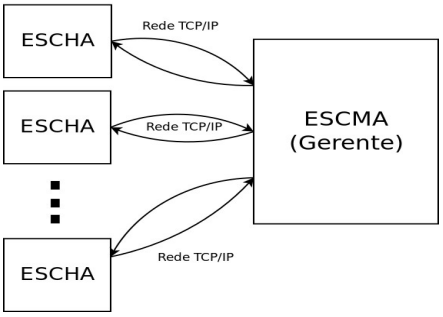

Figura 1. Conexão do hardware remoto com o gerente de alto nível

O dispositivo remoto foi desenvolvido utilizando os microcontroladores AVR de baixo custo da Atmel [1], juntamente com outros periféricos e interfaces que possibilitam a interação com o mesmo. A arquitetura do sistema pode ser dividida em quatro partes, host, interface visual, dispositivo de autenticação (entrada de dados), interface de rede. A figura 2 mostra os componentes envolvidos na comunicação entre um dispositivo de autenticação com a aplicação gerente.

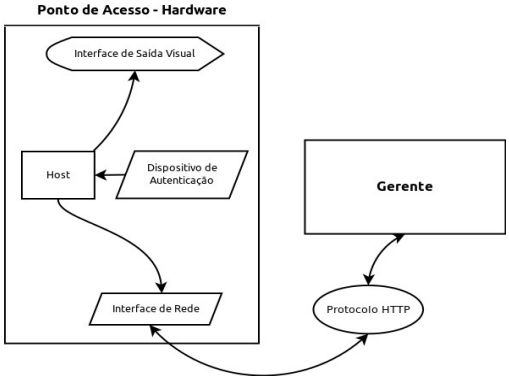

Figura 2. Conexão do hardware remoto com o gerente de alto nível

O dispositivo de hardware desenvolvido consiste em uma interface de coleta de dados e acionamento de outros elementos. Portanto, não é autônomo e depende de um gerenciamento em alto nível para exercer sua função. Sendo esta dependência uma das barreiras que este trabalho se propõe a superar: a dificuldade de comunicação entre um dispositivo de hardware remoto e uma rede de comunicação.

IV. Protocolo de Comunicação

Como foi citado anteriormente, o hardware remoto serve apenas como uma plataforma de coleta de dados e acionamento, sem autonomia própria para autenticação. Para que o sistema funcione efetivamente é necessária uma comunicação com o gerente (ESCMA). Esta comunicação foi desenvolvida na camada de aplicação utilizando-se da já consolidada pilha de protocolos TCP/IP, através do protocolo HTTP. Esta, prove as funcionalidades necessárias ao tráfego de dados entre dispositivos e gerente, utilizando-se, portanto, uma infraestrutura largamente utilizada na construção de qualquer rede de comunicação. Portanto, é eliminada a necessidade da criação de uma nova rede cabeada específica para o dispotivo, como uma rede RS232 para conexão de dispositivos [2]. Outra vantagem do sistema trabalhar na camada TCP/IP é a facilidade de desenvolvimento de softwares que se comuniquem com o dispositivo. Evitando, assim, que o desenvolvedor precise aprender linguagens específicas e possa trabalhar em linguagens nas quais já está acostumado.

Para a comunicação entre dois dispositivos, não basta haver uma canal de comunicação entre eles, também é necessário haver um protocolo de comunicação entre esses elementos.

No âmbito do ESC, utilizou-se como parâmetros para a definição deste protocolos itens básicos que devem ser providos em um sistema de segurança, sendo:

a) Simplicidade: o protocolo de comunicação deve ser implementado, basicamente em dois pontos distintos: a) o ESCMA, onde é implementado através de um linguagem de alto nível, como java e, b) no dispositivo de autenticação, construído baseado na plataforma Arduino, e portanto, com diversas limitações sobre a capacidade de processamento deste. Desta forma, em função de tais limitações, protocolos complexos para comunicação e criptografia não são viáveis.

b) Segurança: o sistema deve, impreterivelmente, utilizar mecanismos que garantam a segurança da comunicação entre o dispositivo de autenticação e gerente. Para isso é essencial a utilização de alguma forma de criptografia.

c) Comunicação Assíncrona: eventos de autenticação são gerados de forma assíncrona, portanto é necessário estabelecer formas de comunicação que permitam minimizar o uso da comunicação da rede, ou seja, o dispositivo deve notificar ao gerente que um evento ocorreu.

d) Independência de dispositivo: independente do dispositivo de hardware utilizado para realizar a autenticação, como RFID, teclado para digitação de senha, etc, o protocolo projetado deve suportar, sem alterações na sua estrutura, as informações necessárias a utilização destes.

Baseado em tais critérios, o protocolo construído é apresentado na figura 3. Seu funcionamento baseia-se em envio de pacotes que seguem um padrão de montagem. O protocolo de comunicação abrange, com poucas modificações no hardware, muitos tipos de entrada de dados, seja ela por digitação de senha, leitura de cartões RFID, leitores biométricos, códigos de barra, etc. A reprogramação deste módulo do dispositivo permite que o desenvolvedor adicione mais métodos de entrada de dados com facilidade.

A. Análise da Transmissão

O pacote de transmissão é montado a partir de vários fatores e dados. Como é visto na figura 3, ele é dividido em várias partes. É necessário notar que pelo fato dos dados serem transmitidos em texto plano, algumas medidas de proteção são necessárias para evitar que o sistema se torne vulnerável à ataques maliciosos, ou até mesmo a problemas de sincronia. Na figura 3 é apresentada uma ilustração que servirá como exemplo para as explicações. Agora, será feita uma análise detalhada da mesma.

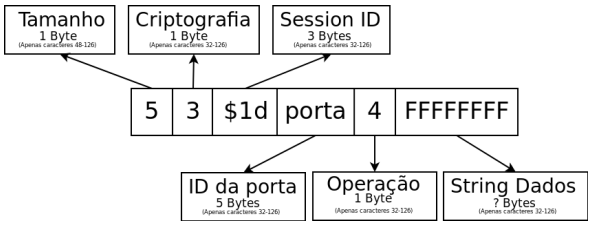

Figura 3. Exemplo de Montagem do pacote de transmissão

1) Tamanho: o primeiro byte da transmissão tem como função evitar erros e leituras incorretas. O tamanho é um inteiro em forma de caracter que representa o tamanho total do pacote da camada de aplicação (construído através de uma string), sua leitura é feita a partir da conversão para valor alfanumérico no padrão ASCII. Só pode assumir valores a partir de '0', ou seja, 48 em alfanumérico. É necessário notar que a string possuí um tamanho mínimo, ou seja, deve receber os blocos principais: tamanho, criptografia, SID, ID da porta e operação, caso contrário ela deve ser descartada, pois não há informações relevantes a serem aproveitadas.

2) Criptografia: este byte irá definir qual o tipo de criptografia a ser usada para desencriptação dos dados relevantes, como os dígitos identificadores de sessão (SID). Os métodos de criptografia devem ser implementados de maneira equivalente entre ESCHA e ESCMA. Métodos de encriptação testados e com bom funcionamento no sistema envolvem embaralhamento e ocultação, algorítmos como o XOR cipher [3] e Caesar Cypher [4]. Devido às restrições de hardware, criptografias unidirecionais (MD5, SHA)[5] não podem ser utilizadas devido ao fato do equipamento não possuir uma base de dados. Esta não é a principal fonte de segurança do sistema, seu principal objetivo é apenas adicionar uma camada de proteção adicional.

3) Session ID: o Session ID, ou dígitos identificadores de sessão, são o principal sistema de segurança e proteção do protocolo de transmissão. Seu princípio de funcionamento é semelhante ao tipo de autenticação "One time password"[6], onde senhas aleatórias são geradas e esquecidas periodicamente. Estes dígitos são gerados sempre que uma comunicação é iniciada, chamada de sessão, e acaba automaticamente após receber uma resposta ou após a passagem de um tempo pré-definido. A SID é criptografada através de algoritmos pré-definidos, e age como uma pergunta que deve ser respondida corretamente para uma transmissão ser bem sucedida. Na figura 4 vemos o gerente recebendo os dados com uma SID criptografada e dados, juntamente com o método de desencriptação a ser usado, logo após receber e processador os dados, uma resposta é gerada (permite/nega acesso, por exemplo) e enviada para o dispositivo remoto, juntamente com a SID desencriptada.

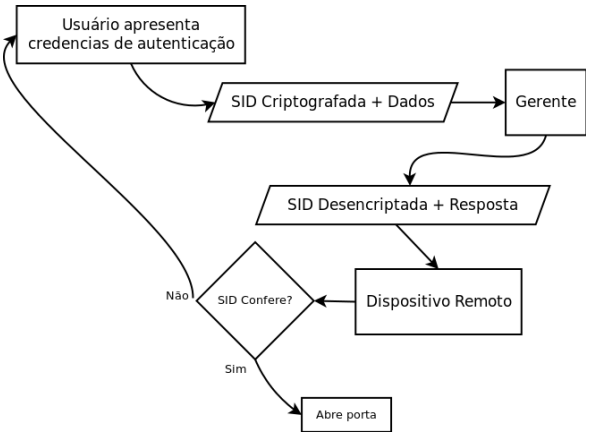

Figura 4. Fluxo de uma comunicação para autenticação

4) ID da Porta: dígitos identificadores da porta servem para informar ao gerente o destino da resposta atual sendo tratada.

5) Dígito de Operação: este byte irá informar aos sistemas qual operação precisa ser feita com os dados passados, exemplos de operações: Abrir porta, permitir ou negar autenticação de dados, enviar mensagem para o display do dispositivo. O sistema está preparado para receber novas implementações de operações sem alterações em sua estrutura principal.

6) String de Dados: este elemento carrega representa informações adicionais a alguma operação. Operações como autenticação podem carregar, por exemplo, dados de um cartão RFID, senhas de um teclado numérico, strings provindas de leitores biométricos entre outros métodos de entrada. Outra função chave do sistema que utiliza os dados é a reposta de autenticação, que contém uma palavra chave informando se a autenticação foi aceita ou negada.

V. Avaliação Experimental

Para análise da solução proposta, montou-se um protótipo integrando o sistema ESCHA ao sistema ESCMA. Para tal, utilizou-se um ambiente de testes alinhado as expectativas de um ambiente real, ou seja, montou-se toda a estrutura de autenticação. Esta, é composta de um dispositivo de autenticação junto aos acessos ao ambiente físico, do hardware necessário para comunicação de rede (ESCHA) interligado, através da rede, ao software para gerenciamento e autenticação centralizada, o ESCMA.

O protótipo do ESCHA foi criado a partir da união de uma placa específica desenvolvida para receber um microcontrolador da Atmel, juntamente com um módulo de interface de rede, que possibilitou a conexão à rede ethernet. Este conjunto está integrado a um módulo de antenas para leitura e escrita em cartões RFID, sendo esta a entrada de dados, e, como a saída de dados, um módulo LCD alfanumérico de 16 colunas e 2 linhas, por final, também foram adicionadas relés para o acionamento de dispositivos externos (fechadura eletromagnética, por exemplo).

O ambiente de teste é ilustrado na figura 5.

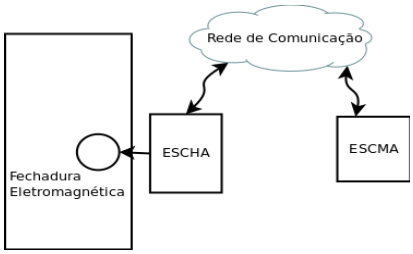

Figura 5. Ambiente de testes ESC

O sistema foi implementado em forma de protótipo para a execução de testes de funcionamento e estresse. Os primeiros experimentos realizados foram estudos para otimizar o funcionamento do protocolo de comunicação, ajustando valores, principalmente de temporização (duração de sessões, tempo entre coleta de dados). O protocolo se mostrou bastante eficiente, dando início aos testes de segurança. Mesmo com a transmissão dos dados ocorrendo sem uma criptografia avançada, em texto plano, a proteção implementada pelo protocolo (criptografia simples e os dígitos identificadores de sessão) dificulta muito a intervenção de programas maliciosos na transmissão. Mesmo com dados analisados através de sniffers, como o Wireshark [7], o processo de quebra da segurança requer níveis exponenciais de estudo e interpretação dos pacotes de pedido e resposta. A primeira vista, os pacotes parecem apenas caracteres aleatórios sem um padrão definido, pois a cada tentativa de acesso uma nova criptografia é usada, juntamente com a criação de uma nova seed que alimenta as mesmas. Apenas um servidor desenvolvido especificamente para aquele hardware será capaz de entender e produzir respostas adequadas para o mesmo.

Com a validação do processo de proteção dos dados, o sistema foi testado sob estresse, forçando requisições de acesso a quantias e intervalos de tempo extremos, quantizando mais de novecentas requisições por hora, em longos períodos de testes. Na tabela 1, é possível observar dados de alguns testes realizados, sendo que no último teste[1] o tempo entre as requisições foi reduzido pela metade, colocando o sistema em uma situação mais extrema, mesmo assim podemos observar uma taxa de falhas de autenticação relativamente baixa.

Período	Tentativas	Sucesso	Falha	%
8 horas	8228	8196	32	99,6
2.5 horas	2812	2807	5	99,8
5 horas	15631[1]	14777	854	94,5

Tabela 1. Testes de estresse executados

O sistema atingiu uma taxa de sucesso acima de 99%, mostrando-se robusto para aplicação prática e pronto para solucionar os problemas que deram início ao mesmo, bem como para receber novas atualizações e gerar, a partir dele, novos e avançados equipamentos de monitoria e controle de ambientes. .

VI. Conclusões e Novos sistemas

Esta é uma arquitetura que pede por atualizações frequentes. As possibilidades são muitas, dentre novas funcionalidades propostas, podemos citar: Mapeamento de pessoas baseando-se no último lugar visitado, sensoriamento e controle de ambientes (como temperatura, luminosidade) através de perfis de usuários, reconhecimento facial e sensoriamento de pessoas num ambiente para controlar outros elementos (travar porta, mapeamento melhorado, controlar aparatos eletrônicos).

Referências

[1] Stanislav Korbel, Vlastimil Janes. Interesting Applications of Atmel AVR Microcontrollers. Department of Computer Science and Engineering of the Czech Technical University. Aug. 2004.

[2] Strangio, C. E. The RS232 Standard. CAMI Research Inc., Acton, Massachusetts (1993) 15.

[3] P. Tuyls, H. D. L. Hollmann, J. H. Van Lint and L. Tolhuizen. XOR-based Visual Cryptography Schemes. Volume 37, número 1. 2005.

[4] Dennis Luciano, Gordon Prichett. “Cryptology” From Caesar Ciphers to Public-Key Cryptosystems. The College Mathematics Journal, Volume 18, Número 1, pp. 2–17. 1987.

[5] Gary C. Kessler, An Overview of Cryptography, Jun. 2010.

[6] N. Haller, C. Metz, P. Nesser, M. Straw. A One-Time Password System. IETF RFC2289, Fev. 1998.

[7] Chris Sanders.Practical Packet Analysis (First ed.). No Starch Press, San Francisco, CA, USA. 2007.

[8] MATTOS, C. L. A. Sentinel: um engenho Java para controle de acesso RBAC. 2003. Pernambuco. Disponível em: <http://www.cin.ufpe.br/~tg/2003-1/clam.doc>. Acesso em: 25/10/2011. Trabalho de Graduação em Segurança da Informação. 50 p.

[9] ID TECH, Soluções. 2006. Disponível em: <http://www.idtech.com.br/solucoes.asp>. Acesso em 08/11/2011.

[10] NIBTEC, Controle de Acesso. 2011. Minas Gerais. Disponível em: <http://nibtec.com.br/produtos.html>. Acesso em 08/11/2011.

[11] MOTTA, Gustavo H. M. B. Um modelo de autorização contextual para ocontrole de acesso ao prontuário eletrônico do paciente emambientes abertos e distribuídos. 2003. São Paulo. Disponível em: <http://www.teses.usp.br/teses/disponiveis/3/3142/tde-05042004-152226/pt-br.php>. Acesso em: 30/8/2012. Tese apresentada à Escola Politécnica da Universidade de São Paulo para obtenção do Título de Doutor em Engenharia.

[1] Teste do sistema, utilizando o intervalo entre requisições reduzido a metade do tempo usual, dos demais testes.