Um Mecanismo de Segurança com Adaptação Dinâmica em Tempo de Execução para Dispositivos Móveis

Alexandre Correia Cirqueira*, Rossana M. C. Andrade, Miguel F. de Castro

Universidade Federal do Ceará (UFC)
Mestrado e Doutorado em Ciência da Computação (MDCC)
GREat – Grupo de Redes de Computadores, Engenharia de Software e Sistemas
Caixa Postal 6001 – 60455-760 – Fortaleza – CE

alexandrecorreia@great.ufc.br, {rossana, miguel}@ufc.br

Abstract. *The increasing use of mobile devices and their applications exchanging information in diverse environments highlights the importance of ensuring data security. Additionally, the trend in the use of sustainable practices advocated by green computing imposes the need for designing applications flexible enough to deal with them. This paper then proposes an adaptive security mechanism, focusing on confidentiality of information exchanged among applications for mobile devices, able to adapt their security degree according to the context and provide the efficient use of resources.*

Resumo. *A crescente utilização de dispositivos móveis e suas aplicações, trafegando informações nos mais variados ambientes, evidenciam a importância da garantia de segurança destes dados. Adicionalmente, a tendência no uso de práticas sustentáveis, defendidas pela computação verde, impõe a necessidade de concepção de aplicações flexíveis o suficiente para lidar com estes desafios. Este trabalho propõe então um mecanismo de segurança adaptativa, com foco na confidencialidade de informações trafegadas por aplicações de dispositivos móveis, capaz de adaptar seu grau de segurança de acordo com o contexto e proporcionar o emprego eficiente dos recursos.*

1. Introdução

A possibilidade de captura de informações trafegadas por aplicações de dispositivos móveis (DMs), com um dispositivo receptor preparado para este fim, aponta para a necessidade de um tratamento eficaz de confidencialidade destas informações [Bringel 2005]. Um mecanismo de segurança que preserva a confidencialidade é a criptografia, a qual atua de forma a tornar os dados compreensíveis apenas para quem possuir sua chave de decifragem [BISHOP, 2003]. Geralmente é empregada em protocolos que oferecem segurança na camada de enlace. Entretanto, como os DMs passeiam por ambientes diversos, nem sempre estes protocolos estão disponíveis ou são adequados para determinadas aplicações que requerem maior segurança. Assim, para garantir o grau de confidencialidade destas informações, [Bringel 2005] sugere que a criptografia deve ser implantada na camada de aplicação e se adequar aos ambientes de uso, pois desta forma, o agente malicioso em posse de algum dado trafegado não encontrará facilidade na tentativa de acesso à informação ali contida.

* Bolsista de mestrado (MDCC/UFC) financiado pela Coordenação de Aperfeiçoamento de Pessoal de Nível Superior CAPES/PROPAG

Por um lado, a manutenção do grau de confidencialidade adequado de uma aplicação, em seus diversos ambientes de uso, possibilita maior segurança em ambientes críticos [Carvalho 2008] e a avaliação de desempenho de algoritmos criptográficos possibilita a construção de mecanismos de segurança eficientes. Por outro lado, o uso eficiente de recursos é um dos princípios fundamentais da computação verde, definida como o estudo e a prática da utilização eficiente dos recursos computacionais, primando pela redução do uso de materiais maléficos à natureza, maximizando a eficiência energética e a vida útil do produto, promovendo sua reciclagem e biodegradabilidade e evitando o aumento dos resíduos de fabricação [Wilbanks 2008].

Sendo assim, neste trabalho defendemos que uma adaptação dinâmica em tempo de execução, alocando adequadamente um algoritmo criptográfico que proporcione o grau de confidencialidade requerido por uma aplicação em cada ambiente de uso, pode garantir a manutenção da segurança e o emprego eficiente dos recursos do DM.

Considerando a construção das aplicações para DMs com a adoção de confidencialidade na forma tradicional, onde um único algoritmo criptográfico é usado de forma estática, o fator de avaliação para sua escolha é um grande desafio, pois este critério de decisão, normalmente baseado no desempenho, é proporcionalmente desfavorável à garantia de confidencialidade proporcionada [Hamad 2009]. Pesquisas de novas técnicas para lidar com o *trade-off* entre o uso de mecanismos de segurança e o consumo de recursos computacionais são destacadas como essenciais para a computação [Maliki 2010], [Taddeo 2010].

Especificamente, o *trade-off* entre prover confidencialidade com o uso adequado de criptografia e diminuir o consumo de recursos dos DMs é a motivação principal deste trabalho. O desenvolvimento de uma solução para engenheiros de software com o intuito de facilitar a adoção da adaptação dinâmica da confidencialidade, em tempo de execução e baseado em informações de contexto, é outra motivação.

A constatação de que uma aplicação nem sempre requer o maior grau de confidencialidade desperta a busca por uma adaptação de acordo com suas necessidades. Para a identificação destas necessidades e indicação da melhor adaptação é necessária a exploração de informações de contexto, o qual é definido como qualquer informação que pode ser usada para caracterizar uma pessoa, lugar ou objeto, que seja relevante na interação do usuário com a aplicação, incluindo ambos [Dey 2001]. A análise do contexto propicia a captura de informações que possibilitam a adaptação eficiente da criptografia, exemplificando, pode-se ter uma aplicação utilizando uma rede classificada como segura, onde ela adapta a criptografia mudando o algoritmo por um que consome menos recursos, poupando o hardware e aumentando sua vida útil.

Aplicações que originalmente requeriam determinado grau de segurança, mas que são construídas com um grau menor (por decisões de negócio), poderiam usar a adaptação dinâmica dos algoritmos de criptografia de acordo com o contexto e restringir a sobrecarga do hardware, ocasionada pelos algoritmos complexos, apenas aos ambientes que necessitam do maior grau de confidencialidade.

Este trabalho evidencia a necessidade de se poupar recursos do DM quando determinadas aplicações permitirem a adaptação de algoritmos criptográficos menos

complexos e oferece alto grau de confidencialidade ao adaptar os algoritmos mais complexos em ambientes de alto risco para aplicações não protegidas adequadamente.

Esta manutenção no nível de confidencialidade aumenta a segurança no tráfego das informações dos usuários de DMs, cada vez mais necessária devido à absorção desta tecnologia no cotidiano dos cidadãos comuns, e também está alinhado com a *green computing* e sua preocupação com o gerenciamento dos recursos.

O restante deste artigo está dividido como segue. A seção 2 apresenta considerações sobre o tratamento de segurança adaptativa em DMs com foco em redução de consumo. Nesta seção também são discutidos os trabalhos relacionados. Na seção 3, a visão geral, os algoritmos e a arquitetura do mecanismo proposto neste trabalho são discutidos. A adoção do mecanismo é tratada na seção 4, onde as atividades necessárias e colaborativas são detalhadas. Na seção 6 são apresentadas as conclusões.

2. Segurança Adaptativa e Eficiência de Consumo

Sistemas de segurança desenvolvidos utilizando os métodos existentes de desenvolvimento de software adaptativo têm sido implementados nos níveis de hardware, do sistema operacional, de rede e no nível de aplicação, porém existe uma consciência geral de que o assunto é relevante e carece de pesquisa [Elkhodary 2007]. Um estudo de metodologias para sistemas de segurança adaptativos e dinâmicos que alteram seu comportamento em tempo de execução apresentado por Elkhodary (2007) indica a necessidade de pesquisas sobre o *trade-off* entre oferta de segurança e eficiência no uso de recursos computacionais, pois ele afirma que a busca pela garantia de segurança normalmente gera sobrecarga do hardware.

Focando nas aplicações de DMs, onde as mudanças de contexto e as necessidades de adaptação podem ser constantes, este trabalho considera as orientações para sistemas adaptativos relacionadas por Elkhodary (2007 apud McKinley 2004), as quais destacam que a segurança adaptativa deve atuar na camada de aplicação e no sistema operacional, e se adaptar em conformidade quando determinados eventos alterarem o nível de ameaça do sistema.

Quanto à definição de complexidade e relevância dos algoritmos criptográficos, há um modelo de avaliação que busca um equilíbrio entre segurança e qualidade de serviço [Chen et al. 2010]. Este modelo considera que os algoritmos criptográficos sejam avaliados pelo tamanho de sua chave (devido à complexidade na avaliação de oferta de segurança) atrelando uma variável que usa pesos de acordo com sua complexidade computacional para definir o nível de garantia de confiança proporcionada. O modelo de Chen (2010) faz uma relação com o impacto no desempenho ocasionado pelo processamento destes algoritmos em dispositivos existentes nas redes sem fio, devido à sobrecarga na alocação de seus recursos.

Este trabalho segue as recomendações citadas nesta seção para a construção do seu mecanismo e relaciona, abaixo, algumas propostas relacionadas. A maioria trata de adaptação de protocolos de segurança na camada de enlace com diferenças bem sutis.

2.1. Trabalhos Relacionados

Os trabalhos mais relevantes encontrados e relacionados com eficiência de consumo de recursos computacionais, quando se trata o quesito segurança, mais especificamente a criptografia de dados para garantia de confidencialidade, estão destacados a seguir.

Um mecanismo de adaptação dinâmica é encontrado em um trabalho [Taddeo 2010] voltado para redes de sensores sem fio (RSSF), o qual adapta a segurança de forma gradual pela seleção de algoritmos criptográficos. Segundo os autores, normalmente se utiliza uma abordagem onde, ou se usa um algoritmo de segurança ou não se usa nenhum, a qual se estende para os ambientes de computação móvel, contrariando as exigências de requisitos para estes dispositivos, que incluem mobilidade, flexibilidade, configuração em tempo real e dinamicidade de ambientes, pois o mesmo algoritmo é usado em todos os ambientes percorridos. O trabalho destaca também que um dos tópicos mais relevantes na atualidade, e também o problema principal e complexo neste tipo de adaptação, é a manutenção do nível adequado de proteção, principalmente em tempo de execução. Os autores propõem então um mecanismo que lida com adaptação de segurança de acordo com os requisitos de segurança da aplicação e com as restrições de energia de forma dinâmica. Baseiam-se no princípio da proteção adequada, o qual prega que uma informação deve ser protegida em escalas consistentes com este valor, sendo assim aplicada adequadamente de acordo com o contexto [Taddeo apud Pfleeger 2006]. Seu foco principal é satisfazer os requisitos de consumo de energia, atuando na adaptação do nível do sinal propagado, encerrando algumas aplicações. Ele provê um algoritmo para todas as aplicações em uso enquanto o MeSAD mantém o algoritmo adequado para cada aplicação e ainda provê uma avaliação para melhor prover esta adaptação baseada no contexto, mas sem encerrá-las como a abordagem de Taddeo (2010) o faz.

Outro trabalho utiliza a adaptação com base em estudos de possíveis cenários para decisões em tempo de projeto [Hamad et al. 2009], considerando sua adoção em dispositivos com limitações de recursos. Ele investiga alguns algoritmos, considerados pelos autores como os mais populares quanto às suas complexidades e utilização de recursos. Sua adaptação está centrada em oferecer aos usuários a opção de realizarem escolhas do melhor esquema de segurança a adotar de acordo com essas informações. Difere da proposta aqui tratada porque trata as adaptações em tempo de projeto, de forma não dinâmica e não age em tempo de execução.

Um serviço de criptografia adaptativo é proposto por Izquierdo et al. (2007), levando em conta as limitações de capacidade de processamento e requisitos de segurança. É uma arquitetura de criptografia de dados que usa mais de um algoritmo para criptografar vários blocos de informação. O serviço usa a criptografia de apenas alguns blocos da informação, ou seja, dentre o total dos blocos trafegados alguns são enviados sem proteção, também pode usar outros algoritmos em paralelo para criptografar diferentes blocos. A diferença do presente trabalho deve-se ao tratamento de segurança a ser aplicado no MeSAD considerar a criptografia da informação por completo sem dividí-la em blocos, e também por não deixar trechos da informação trafegar sem proteção, o que Izquierdo (2007) afirma que acontece.

Com foco em desempenho, o trabalho de Rocha et al. (2007) é um middleware adaptativo de seleção dinâmica de protocolos de segurança na camada de enlace, que

usa variações de parâmetros da rede sem fio, recursos disponíveis do sistema em uso e níveis de segurança. Trata diferentes tipos de dados com configurações específicas de segurança. Difere deste trabalho por focar na camada de enlace, enquanto o mecanismo proposto neste artigo, apesar de focar na camada de aplicação, atua transversalmente, adaptando seu grau de confidencialidade.

O Prometheus [Pirmez 2009] é definido como um serviço de segurança ciente de contexto capaz de adicionar controles de segurança dinamicamente. Não oferece algumas características destacadas como importantes pelo próprio trabalho, como considerações sobre medições de consumo de energia e o devido tratamento para redução de consumo. Sua adoção está focada em nichos de aplicações onde as políticas de utilização devem estar bem definidas, com regras de execução entre si e com definições de prioridades de execução, diferentemente deste trabalho, que é caracterizado pela necessidade de independência das aplicações em execução e não limitação de seu uso, e que tampouco encerra qualquer aplicação em uso.

O SARM (Security Adaptation Reference Monitor) [Maliki 2010] foi proposto para lidar com ambientes dinâmicos das redes sem fio e também defende a abordagem de formas flexíveis para lidar com a segurança de maneira dinâmica, em tempo de execução e com uso de informações de contexto. Propõe um monitor genérico de adaptação de segurança considerando o desempenho em relação ao consumo de energia. Também defende o uso de adaptação de segurança em tempo de execução na rede sem fio para atenuar as conseqüências do número substancial de ameaças quando não se pode eliminá-las por completo, pois não há consciência de qual mecanismo de segurança se deve utilizar nas aplicações e que sejam tão dinâmicos na proteção quanto são as ameaças. Para cada aplicação, o usuário deve preencher uma avaliação técnica detalhando suas preferências de uso e é baseado nestas avaliações que o SARM realiza as adaptações. Restringe-se a avaliar ou não o uso de segurança e, com esta decisão, considera que está economizando energia. Diferencia-se deste trabalho pela maneira como lida com as informações de contexto, por não tratar escalas de segurança, não levantar informações de consumo de energia e concentrar as avaliações de todas as aplicações no monitor a partir do kernel.

Tabela 1. Principais características dos trabalhos relacionados e o MeSAD

Abordagens propostas	Usa graus ou escalas de segurança	Sensível ao Contexto	Trata economia de recursos	Mantém sempre um nível de segurança	Adaptação dinâmica em tempo de execução	Atua na camada de aplicação	Mantém independência das aplicações
Taddeo	✓	✓	✓		✓		
Hamad	✓		✓	✓			
Izquierdo		✓	✓		✓		
Rocha	✓	✓	✓		✓		
Pirmez		✓		✓	✓	✓	
Maliki		✓	✓		✓		
MeSAD	✓	✓	✓	✓	✓	✓	✓

Na Tabela 1 pode ser verificado que os outros trabalhos deixam de fora pelo menos um dentre os aspectos que são considerados no MeSAD, o qual respeita os requisitos de segurança e de execução de cada aplicação, usando a sensibilidade ao contexto na adaptação dinâmica e trata a eficiência na utilização dos recursos do DM.

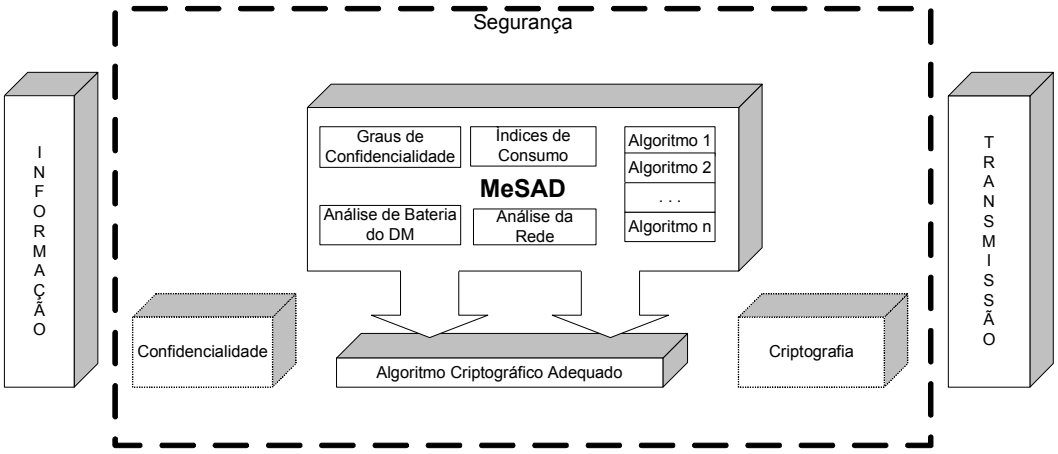

Figura 1. Visão Geral do MeSAD

3. Mecanismo de Segurança com Adaptação Dinâmica

A confidencialidade tratada na camada de aplicação tem a vantagem da exploração das propriedades específicas das aplicações para possibilitar o emprego mais eficiente dos mecanismos de segurança. Assim, o Mecanismo de Segurança com Adaptação Dinâmica (MeSAD) proposto neste trabalho, visa explorar as características individuais das aplicações para melhor adaptar o tipo de criptografia necessária, diminuindo a carga imposta pelo custo computacional desta operação e preservando os recursos do DM.

3.1. Visão Geral

Uma visão geral do funcionamento do MeSAD pode ser visto na Figura 1, onde uma aplicação que precisa garantir confidencialidade da Informação em tempo de execução, implanta aspectos de Segurança para realizar a Transmissão de seus dados. O requisito de segurança considerado é o de Confidencialidade que utiliza a Criptografia dos dados através de um Algoritmo Criptográfico. Este último é definido pelo MeSAD que utiliza informações de contexto para avaliar qual é o algoritmo mais adequado.

O ambiente no qual são utilizados os DMs apresenta uma série de particularidades, dentre elas a necessidade da manutenção de diferentes níveis de confidencialidade para os diversos graus de riscos de utilização deste dispositivo. O MeSAD foi projetado para resolver este problema mantendo o nível de segurança adequado, de maneira dinâmica, à medida que o DM passeia pelos diversos ambientes.

Neste tipo de cenário, um aspecto crítico de sistemas preocupados com contexto é a capacidade de adaptar seu comportamento de acordo com o contexto atual e, para isso, informações devem ser adquiridas de fontes externas (e.g., sensoriamento). O MeSAD utiliza informações de contexto capturadas em tempo de execução para definir a criptografia mais adequada a fim de garantir o grau adequado de confidencialidade da aplicação. Desta forma, ele possibilita que diferentes aplicações possam utilizar algoritmos de criptografia distintos de acordo com seu contexto e classificação de grau de confidencialidade, proporcionando o emprego mais eficiente dos recursos e evitando a carga proporcionada pelas soluções de segurança generalizadas.

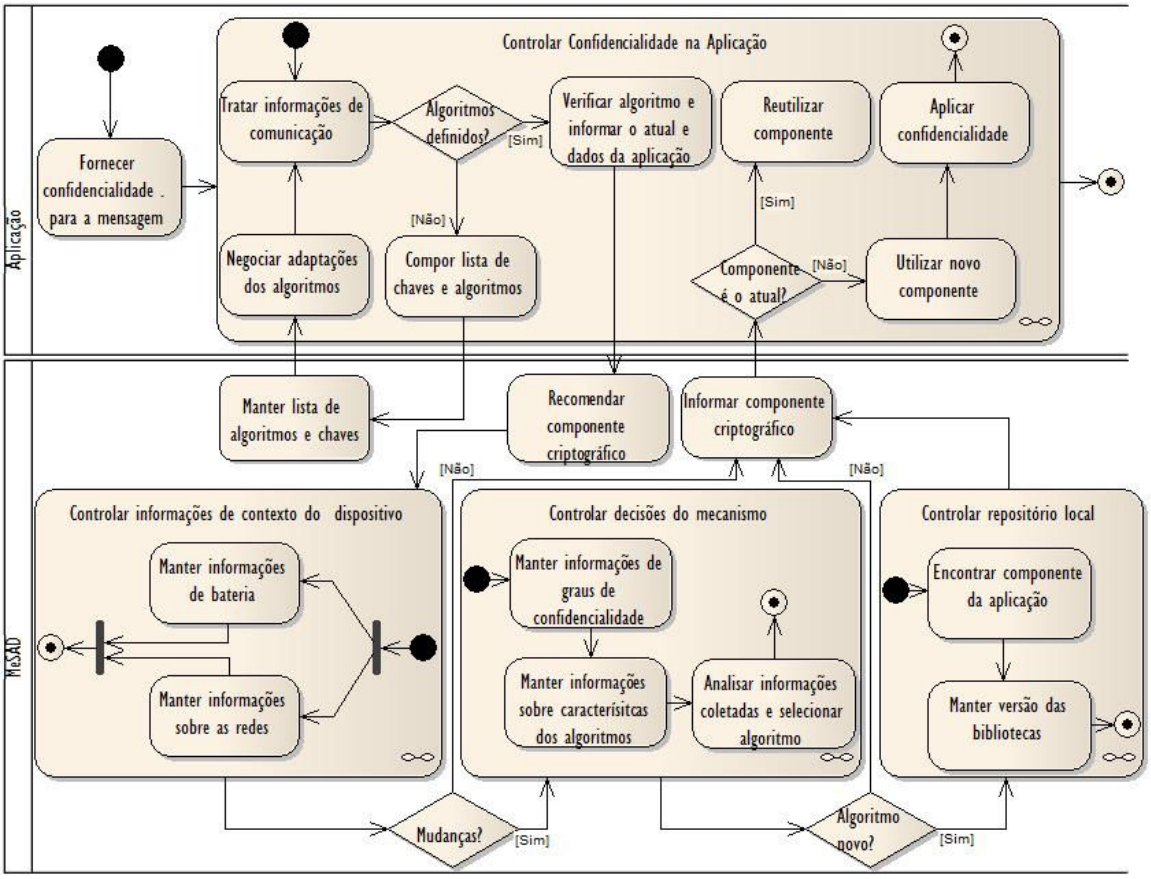

Figura 2. Os algoritmos do MeSAD dispostos na aplicação e no DM

Na Figura 2 estão evidenciados os algoritmos do mecanismo proposto, onde o módulo criptográfico que estará embutido na aplicação se encontra detalhado na raia **Aplicação** e o algoritmo do MeSAD em funcionamento no DM está detalhado na raia **MeSAD** da figura. A figura mostra a atividade de fornecimento de confidencialidade para a mensagem, iniciada pela aplicação e a partir dali as atividades do MeSAD interagem para controlar a escolha mais acertada do algoritmo e a utilização do componente nesta oferta de segurança criptográfica para a mensagem.

Para proporcionar segurança individualizada por aplicação e adaptativa, o MeSAD faz uso de informações de configuração da aplicação, informações de contexto capturadas em tempo de execução, histórico sobre redes utilizadas pelo usuário e classificações configuradas em tempo de projeto. As informações de contexto obtidas em tempo de execução, juntamente com as informações geradas em tempo de projeto irão influenciar a tomada de decisão.

O funcionamento do MeSAD é demonstrado na Figura 3, onde um repositório local conterá todos os componentes necessários para as aplicações instaladas no DM e todos os componentes implementam as interfaces que possibilitam que as aplicações os utilizem. A aplicação irá utilizar o módulo MeSAD UsaCripto, existente em seu Módulo Criptográfico local e responsável por tratar a criptografia e descriptografia das suas informações. Ele possui um componente que implementa a interface da biblioteca de

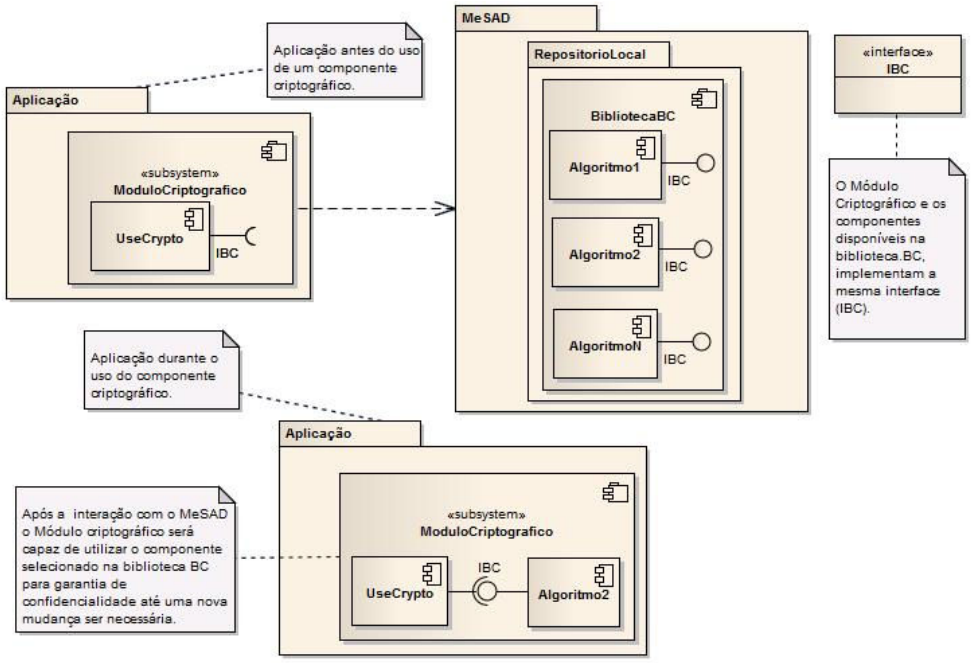

Figura 3. Uma aplicação sendo provida por um componente de criptografia

segurança adotada representada por IBC. Quando necessitar utilizar criptografia, ele irá consultar o módulo principal MeSAD existente no DM, passando informações da aplicação. O mecanismo analisará as informações de contexto e de configuração, aplicará suas regras de decisão e acessará seu Repositório Local para localizar a biblioteca do componente identificado e o fornecerá para que UseCrypto o utilize.

No caso representado pela Figura 3, o componente implementa a interface da biblioteca BC e o MeSAD está escolhendo um componente desta biblioteca, a qual implementa a interface IBC que provê compatibilidade com UseCrypto. Estes componentes poderão, em tempo de execução, serem substituídos de acordo com a necessidade da aplicação e com o contexto identificado no momento do uso.

3.2. Arquitetura do MeSAD

O MeSAD possui uma arquitetura focada no uso de componentes para prover adaptação dinâmica, pois eles podem ser adicionados, substituídos e reconfigurados em tempo de execução, permitindo adaptação às novas necessidades [Rocha 2007 apud McKinley 2004]. O MeSAD é instalado no DM e os módulos criptográficos (subsistemas) das aplicações que o adotam realizam a interação, sendo assim é dividido em duas partes: a solicitante e a fornecedora.

Na Figura 4, pode-se observar a existência de um Módulo Criptográfico na representação da Aplicação. Ele é adicionado na implementação das aplicações que utilizam o mecanismo e é responsável por acionar o MeSAD, provendo informações sobre versão e configuração da aplicação, e por tratar a confidencialidade. O pacote MeSAD representa a maior parte do mecanismo instalado no DM e é responsável por atender às requisições de decisão e fornecer os componentes para as aplicações. Ele aciona o Módulo de Decisão ao receber as informações sobre a aplicação que requer

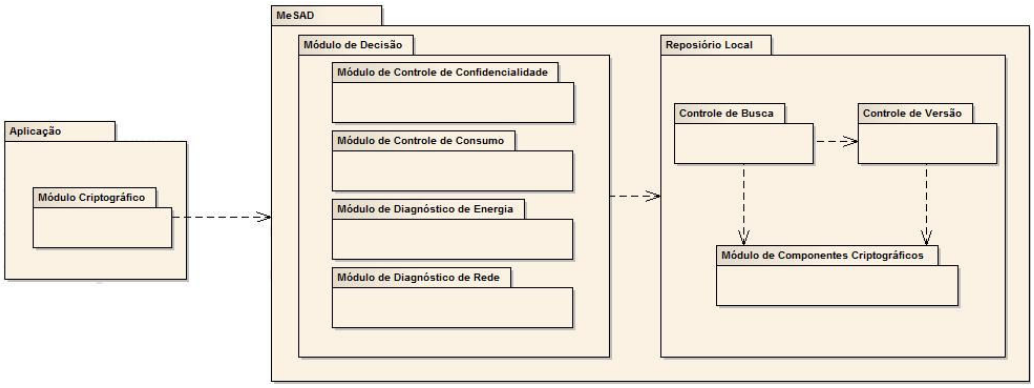

Figura 4. Diagrama de pacotes do MeSAD

criptografia. Este módulo utiliza seus mecanismos (representados por seus módulos internos) para capturar as informações necessárias para a tomada de decisão sobre qual algoritmo de criptografia é o mais adequado para a situação verificada e, em seguida, consulta o Repositório Local para encontrar o componente que o representa. Este último analisa a informação e verifica a versão e o local de busca para suprir a aplicação.

A seleção dos algoritmos candidatos será possível com o uso dos graus de confidencialidade, a análise de confiança das redes, o nível de bateria do DM e o índice de consumo de energia dos algoritmos.

Com o algoritmo criptográfico definido, ele será utilizado pela aplicação até que o mecanismo identifique a necessidade de uma nova adaptação. Esta mudança ocorrerá caso o nível de bateria entre em estado crítico, a rede utilizada mude ou o mecanismo solicite a mudança divulgando as novas configurações pelo protocolo da aplicação.

Todo o processo de uso e substituição dos algoritmos está ilustrado na Figura 5, onde estão representados CtrlCripto e UseCripto, que implementam as interfaces do MeSAD e são partes do Módulo Criptográfico presente em cada aplicação. O CtrlCripto controla o acesso ao MeSAD para obter o componente de criptografia mais adequado para a aplicação, de acordo com o contexto, e o fornece ao UseCripto, que, por sua vez, coloca em prática a cifragem e a decifragem das informações. Ele também verifica se o componente precisa ser modificado e, em caso positivo, fornece o novo componente para UseCripto, estando representado na figura pelo laço e a condição de mudança do componente. Pode-se notar a interação do Módulo de Decisão com seus módulos internos que representam as atividades do mecanismo (graus de confidencialidade, índices de consumo, análise de bateria e análise de rede) e também o acesso ao Repositório Local onde será identificado o componente de acordo com as necessidades e configurações da aplicação.

A parte do MeSAD que permanece na aplicação e trata a confidencialidade das mensagens está descrita no diagrama da Figura 6, o qual mostra a interação do Módulo Criptográfico da aplicação com os outros módulos. A aplicação interage com CtrlCrypto passando o texto para que ele promova o tratamento (cifragem/decifragem). Informações de configuração desta aplicação são colhidas e enviadas ao MeSAD do DM para fornecimento do componente de criptografia adequado. Em seguida, informações como chaves, algoritmo inicial e parâmetros para mudanças posteriores são protegidas

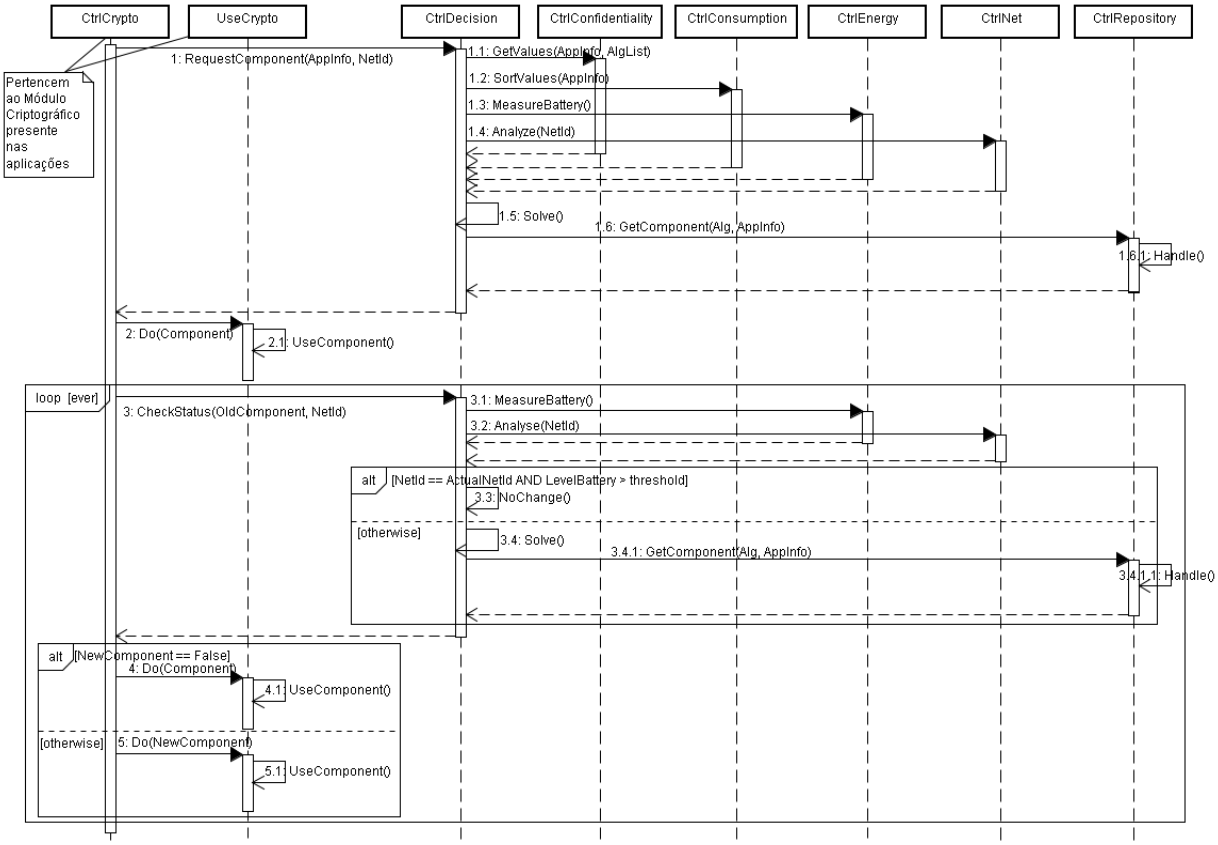

Figura 5. Diagrama de Seqüência Simplificado do MeSAD

com criptografia assimétrica e um protocolo de handshake inicia a comunicação para que seja estabelecido o mesmo algoritmo entre as partes (CtrlProtocolo). Após esta definição a mensagem é criptografada com o algoritmo simétrico estabelecido e enviada. Quando for necessário nova adaptação, esta informação será adicionada na última mensagem criptografada com o algoritmo anteriormente definido para divulgação do próximo algoritmo a ser utilizado nas próximas mensagens trocadas.

Quanto à decifragem de uma mensagem recebida, a Figura 6 mostra a interação do protocolo para verificação da lista de chaves e a verificação do componente em uso na criptografia da mensagem recebida. Caso ele já esteja em uso não é necessário novo fornecimento, caso contrário o MeSAD fornecerá o novo componente para uso.

4. Implantação do MeSAD

A adoção do MeSAD em um projeto de software adiciona atividades que vão desde a fase de concepção até a manutenção da aplicação, seguindo as orientações de Elkhodary (2007). Incorpora a preocupação com segurança nos processos de criação de softwares, tão necessária e cada vez mais requisitada, porém tão pouco utilizada [Chen et al. 2010].

Em cada fase do ciclo de vida do software, há uma maneira de atuar para a geração dos artefatos necessários para a utilização eficaz do mecanismo proposto. Porém, na fase de projeto, uma vez definidas as configurações de níveis de

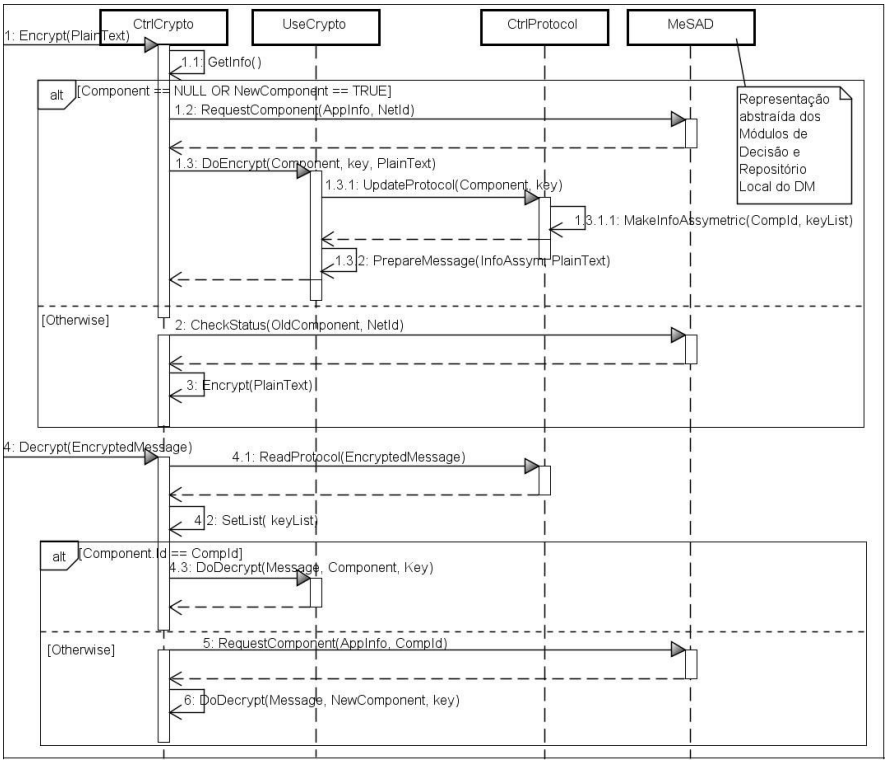

Figura 6. Diagrama de Seqüência da parte do MeSAD presente na aplicação

confidencialidade a ser praticada nas aplicações que utilizarão o MeSAD, esta etapa do processo poderá ser reutilizada para todas as novas aplicações, facilitando sua adoção.

Nas fases de análise e de projeto são definidos ou reutilizados os graus de confidencialidade pertinentes à aplicação que será criada e isto acontecerá de acordo com os requisitos e arquitetura avaliados no projeto. Também serão criados ou reutilizados, os índices de consumo de bateria para cada algoritmo criptográfico. No momento da construção do código as informações dos graus de confidencialidade serão utilizadas para a geração e introdução dos arquivos do MeSAD na aplicação. Os testes indicarão a possibilidade de refinamento dos graus definidos como entradas do mecanismo. A arquitetura do mecanismo facilita o processo de evolução ou manutenção da aplicação, pois basta definir os novos graus de confidencialidade e gerar as novas entradas do mecanismo para serem substituídas na aplicação.

4.1. Atividades Colaborativas

O funcionamento do mecanismo está atrelado às atividades que colaboram entre si, adquirindo os valores de entrada para as tomadas de decisões e execução das ações.

4.1.1. Graus de Confidencialidade

Como citado anteriormente, existem informações que deverão ser geradas em tempo de projeto e elas são fundamentais para o correto funcionamento do mecanismo. Dentre elas, a definição dos **graus de confidencialidade dos algoritmos de criptografia** contidos nas bibliotecas de segurança e a definição do **grau de confidencialidade da aplicação** que estará sendo criada de acordo com seus requisitos de segurança.

No projeto da aplicação, o engenheiro de software fará uso de uma classificação de grau de confidencialidade ofertada pelos algoritmos de criptografia. Esta classificação definida no MeSAD é exposta e justificada nos parágrafos seguintes.

O nível de garantia de segurança proporcionado pela criptografia é determinado pelo algoritmo criptográfico e pelo tamanho de chave utilizada [Chen 2010]. Assim, a classificação de graus de confidencialidade dos algoritmos se baseou neste argumento.

Analisando a possibilidade de tentativas de quebra de segurança por "força bruta" e considerando que um comprimento de chave com 8 bits ($2^8 = 256$) possui 256 possibilidades de definição de chaves, ou seja, no pior caso é preciso tentar 256 vezes até se descobrir a chave, substituindo o expoente por uma variável k de tamanho igual à chave utilizada (k_{tam}), pode se generalizar esta relação como: $2^{k_{tam}}$ **(1)**.

A expressão **(1)** representa o número de tentativas para obtenção da chave de criptografia e, assim definido, pode-se concluir que quanto maior for o k_{tam}, maior será o nível de segurança proporcionado por um algoritmo. Assim, para classificar os graus de confidencialidade proporcionados, assume-se que o valor deste grau é 1 quando for utilizado o menor tamanho de chave (k_{min}). Então, o grau de confidencialidade da criptografia (l_k) pode ser definido como: $\mathbf{l_k} = 2^{k_{tam}/k_{min}} - 1$ **(2)**.

Definido este formato, deve-se agora considerar a complexidade da criptografia oferecida pelos variados algoritmos criptográficos. Como na literatura não há uma definição concisa de uma classificação de cada algoritmo de acordo com sua complexidade e importância no processo de manutenção da confidencialidade com base em seus poderes computacionais, pode-se definir as diferenças entre os algoritmos a serem considerados pela atribuição de um peso (p), com valores de 0 até 1, na equação **(2).** Desta forma, tem-se agora: $\mathbf{l_k} = (2^{k_{tam}/k_{min}} - 1) * p$ **(3)**.

A atribuição do peso (p) facilita a adoção do MeSAD por engenheiros de software que queiram modificar suas configurações originais de acordo com seus conhecimentos e necessidades, além de oferecer a facilidade de novas classificações em futuras versões do mecanismo, com a inclusão de novos algoritmos.

Com os graus de confidencialidade dos algoritmos de criptografia definidos, é necessário definir o grau de confidencialidade da aplicação que está sendo criada. Assim, o engenheiro de software deve identificar a necessidade de segurança que sua aplicação requer e classificá-la nas opções disponíveis na ferramenta de suporte ao MeSAD. Ela segue o modelo comumente adotado em aplicações que utilizam classificações de níveis de segurança e no trabalho adotado para estas definições mencionadas acima [Chen 2010], o qual também utiliza três níveis. São eles: baixo, médio e alto. Um destes níveis será atribuído à aplicação para a determinação de seu requisito de segurança de acordo com suas necessidades e para a composição dos parâmetros de configuração nos arquivos que o mecanismo produzirá.

4.1.2. Índice de Consumo do Algoritmo de Criptografia

O engenheiro de software deve produzir ou reutilizar testes para avaliar o impacto do uso de cada algoritmo criptográfico quanto ao consumo de energia da bateria dos DMs, ou seja, os dispositivos nos quais se espera que a sua aplicação esteja em uso poderão

ser utilizados em testes para produzir uma lista de índices de consumo de bateria, possibilitando a criação de uma média de consumo para determinado categoria de DMs.

Estes testes possibilitarão a criação de outra entrada do MeSAD. Estas informações serão obtidas pelo uso de uma ferramenta de apoio com este intuito. Este software está sendo criado para ser uma ferramenta de coleta e análise de informações relacionadas com a execução de diversos algoritmos criptográficos em DMs, tratando-se de um avaliador de algoritmos criptográficos com relação ao consumo de energia da bateria por eles proporcionado. Para adquirir esta informação de índices de consumo, funcionará basicamente da seguinte forma: uma aplicação é instalada nos DMs que irão ser testados, alguns parâmetros são informados e a aplicação é então executada. Os resultados dos testes são enviados para um servidor web e mantidos para a criação de entradas de configuração do mecanismo a ser utilizado.

4.1.3. Análise de Confiança

Como confiança é uma sensação que provém do indivíduo, o qual interage com os elementos que propiciam ou não tal sensação, sua participação na definição de um valor que reflita o grau de confiança nas redes em uso é fundamental. Assim sendo, o MeSAD dará suporte para o usuário realizar uma avaliação de suas redes e irá considerá-la para conceber um grau de confiança em cada rede onde utilizará criptografia.

Esta informação possui duas formas de ser identificada se complementam para estabelecer qual rede é mais ou menos confiável. Uma delas é através de uma análise que é feita no momento do uso da aplicação, onde a rede em atividade é avaliada para identificar os mecanismos que incrementam a segurança no tráfego via rádio das informações, caso existam. A segunda forma de identificar o grau de confiança da rede é a análise direta do usuário quanto à sua confiança na rede que ele está utilizando. Esta avaliação é importante porque apenas o usuário pode informar o quanto ele confia ou não em determinadas redes e de acordo com as características desta rede, este usuário pode avaliá-la e definir o seu nível de confiança, o qual será mantido para reutilização. Um exemplo de como esta informação é restrita ao conhecimento do usuário seria uma rede existente num ambiente remoto ou isolado (e.g., uma chácara com acesso à internet fornecido por um ponto de acesso de fraca segurança em seu enlace aéreo), onde o usuário sabe que não existiria um agente malicioso nas proximidades capaz de capturar a sua transmissão. Outro exemplo seria o usuário acessando uma rede a partir de um aeroporto em um ponto de acesso com tecnologia moderna de proteção, porém com chave de acesso compartilhada por várias pessoas, diminuindo sua confiança nesta rede.

4.1.4. Análise do Nível de Bateria

Como o MeSAD atua em tempo de execução para propiciar a adaptação dinâmica, de acordo com informações de contexto e sendo uma destas informações ligada diretamente com a preocupação de consumo eficaz de recursos computacionais com foco na bateria, nada mais natural que o nível de energia do DM seja monitorado para refinar a escolha do algoritmo criptográfico para uso na garantia de confidencialidade. Esta informação é capturada em determinados instantes, quando a adaptação estiver sendo considerada e pode influenciar na escolha de um algoritmo que consome menos de acordo com o nível da bateria.

5. Conclusões

Este trabalho apresentou o MeSAD, cujo objetivo é proporcionar a segurança adequada para aplicações de DMs em ambientes considerados inseguros, e diminuir a saturação no uso de recursos quando estiverem em ambientes considerados seguros, através da adaptação da confidencialidade em tempo de execução. A construção do MeSAD considerando questões como computação verde, aumento de confidencialidade em ambientes críticos, adaptação em tempo de execução de acordo com o contexto e tratamento individualizado de confidencialidade por aplicação, o coloca em sintonia com os desafios do futuro da computação.

Para facilitar a implantação do MeSAD nas aplicações criadas para DMs, uma ferramenta de apoio está em desenvolvimento e será responsável por configurar informações sobre a aplicação, avaliar o consumo de energia dos algoritmos criptográficos e gerar os arquivos necessários para a utilização do mecanismo. Diminuindo consideravelmente o trabalho do engenheiro de software que precisa tratar a criptografia das informações nas suas aplicações.

No futuro, pretende-se ainda disponibilizar o maior número possível de bibliotecas de segurança e adicionar outros requisitos de segurança, além da confidencialidade, bem como oferecer o mecanismo em diferentes plataformas.

Referências

Bishop, M., Computer Security: Art and Science, Addison-Wesley, 2003.

Bosch, Jan. Design and Use of Software Architecture, Addison-Wesley, 2000

Bragg, R., et al. (2004). Network Security: The Complete Reference. McGraw-Hill.

Bringel, J. R. M. F., Viana, W., Braga, R., Andrade, R. M. C. (2005).FRAMESEC: A Framework for the Application Development with End-to-End Security Provision in the Mobile Computing Environment. In: AICT 2005. Proc. IEEE Computer Society.

Carvalho, A. F. M. (2008). M-CODE: Um Modelo para Medição de Confidencialidade e Desempenho para Aplicações Móveis Seguras. 96p. Dissertação (Mestrado em Ciência da Computação) – Universidade Federal do Ceará, Brasil.

Chen, J., Hu, C., Zeng, H., (2010). A Novel Model for Evaluating Optimal Parameters of Security and Quality of Service. Journal of Computers, vol 5, no. 6.

Dey, A.K. (2001). Understanding and Using Context. Personal Ubiq. Comp. v5, pp4-7

Darco, P., DE Santis, A., Feara, A. L., Massucci, B. (2010). Variations on a theme by Akl and Taylor: Security and tradeoffs. Theor. Comput. 411, 1, 213-227.

Elkhodary, A., Whittle, J. (2007). Survey of Approaches to Adaptive Application Security. ICSE Workshops SEAMS '07. pp. 16-16.

Gentry, C., Ramzan, Z. (2004). Provable cryptographic security and its applications to mobil wireless computing. Wireless Personal Communication, v. 29, pp.191-203.

Georgas, J.C., Hoek A.V.D., Taylor R.N. (2005). Architectural Runtime Configuration Management in Support of Dependable Self-Adaptive Software, WADS, v30,pp. 1-6.

Hamad, F., Smalov L., James A. (2009). Energy-aware Security in M-Commerce and the Internet of Things. IETE. 26:357-62.

Herrick, D. R., Ritschard, M. R. (2009). Greening your computing technology, the near and far perspectives. In Proceedings of the ACM SIGUCCS Fall Conference on User Services Conference. SIGUCCS '09. ACM, New York, NY, 297-304.

Izquierdo, A., Sierra, J.M., Torres, J. (2006). On the implementation of security policies with adaptative encryption. Comput. Commun. 29(15), 2750–2758.

Li, H.(2006).Multilievel Adaptive Security System.92p. Thesis(Doctor of Philosophy in Computer Engineering).New Jersey Institute of Technology(NJIT),New Jersey, USA

Maliki, T.E.; Seigner, J., (2010). A Security Adaptation Reference Monitor (SARM) for Highly Dynamic Wireless Environments. Emerging Security Information Systems and Technologies. International Conference, pp.63-68, 18-25.

Oreiz, P., et. al(2008).Runtime Software Adaptation: Frameworks, Approach and Styles. 30th International Conference on Software Engineering. ICSE'08. ACM, pp 899-910.

Pirmez, M. (2009). Prometheus: Um Serviço de Segurança Adaptativa. 115p. Dissertação (Mestrado em Informática) - Universidade Federal do Rio de Janeiro.RJ.

Rocha, B.P.S., Costa, D.N.O., Moreira,(2010).Adaptive security protocol selection for mobile computing. J. Netw. Comput. Appl. 33, 5 (September), 569-587.

Rocha, L.S., Castro, C. E. P. L., Machado, J. C., Andrade, R. M. C. (2007). Utilizando Reconfiguração Dinâmica e Notificação de Contextos para o Desenvolvimento de Software Ubíquo. In: XXI SBES. João Pessoa.

Salido, J., Lazos, L., Poovendran R. (2008). Energy and bandwidth-efficient key distribution in wireless Ad Hoc networks: A cross-layer approach. IEEE/ACM Transactions on Networking, vol. 15, No. 6, pp. 1527-1540. DEC.

Schilit, B., Adams, N., Want, R.(1994). Context-aware computing applications. In IEEE Workshop on Mobile Comp. Systems Applications. IEEE Comp. Soc. Press. p85-90.

Solyman, S. H., Omari M. (2004). An Efficient Application of a Dynamic Crypto System in Mobile Wireless Security. WCNC/IEEE C. Society, vol.2, pp. 837-842.

Stephen, Ruth. (2009). Green IT – More Than a Three Percent Solution, IEEE Internet Computing, vol. 13, no. 4, pp. 74-78, July/Aug., doi:10.1109/MIC.2009.82.

Szyperski, C. A. (1998). Component software: beyond object-oriented programming, ACM Press/Addison-Wesley Publishing Co., New York, NY.

Taddeo, A.V. et al.(2010).Gradual Adaptation of Security for Sensor Networks. In:IEEE International Symposium on a World of Wireless Mobile and Multimedia Networks.

Viana, W., Cavalcante, P., Andrade, R. M. C. (2005).Mobile Adapter: Uma abordagem para a construção de Mobile Application Servers adaptativos utilizando as especificações CC/PP e UAProf. In: XXV Congresso da Sociedade Brasileira de Computação, 2005, Sao Leopoldo, RS-Brasil. Anais do XXXII SEMISH.

Wilbanks, L. (2008). Green, My Favorite Color. IT Pro v.10, n.6. pp.63-64. IEEE Comp Soc. Press.