Ministério da Justiça

Termo de Cooperação/Projeto:

Acordo de Cooperação Técnica FUB/CDT e MJ/SE Registro de Identidade Civil – Replanejamento e Novo Projeto Piloto

Documento:

RT Diagnóstico sobre eIDs e Pesquisa de Tecnologias

Parte III: Estudo de Protocolos de Autenticação em Modelos de eID

Data de Emissão:

13/09/2015

Elaborado por:

Universidade de Brasília – UnB
Centro de Apoio ao Desenvolvimento Tecnológico – CDT
Laboratório de Tecnologias da Tomada de Decisão – LATITUDE.UnB

MINISTÉRIO DA JUSTIÇA | **UNIVERSIDADE DE BRASÍLIA**

José Eduardo Cardozo
Ministro

Ivan Marques Toledo Camargo
Reitor

Marivaldo de Castro Pereira
Secretário Executivo

Paulo Anselmo Ziani Suarez
Diretor do Centro de Apoio ao Desenvolvimento Tecnológico – CDT

Helvio Pereira Peixoto
Coordenador Suplente do Comitê Gestor do SINRIC

Rafael Timóteo de Sousa Júnior
Coordenador do Laboratório de Tecnologias da Tomada de Decisão – LATITUDE

EQUIPE TÉCNICA

Ana Maria da Consolação Gomes Lindgren
Andréa Benoliel de Lima
Celso Pereira Salgado
Delluiz Simões de Brito
Elaine Fabiano Tocantins
Fernando Saliba Oliveira
Fernando Teodoro Filho
Guilherme Braz Carneiro
Joaquim de Oliveira Machado
José Alberto Sousa Torres
Marcelo Martins Villar
Raphael Fernandes de Magalhães Pimenta
Rodrigo Borges Nogueira
Rodrigo Gurgel Fernandes Távora
Sara Lais Rahal Lenharo

EQUIPE TÉCNICA

Flávio Elias Gomes de Deus (Pesquisador Sênior)
William Ferreira Giozza (Pesquisador Sênior)
Ademir Agustinho de Rezende Lourenço
Adriana Nunes Pinheiro
Alysson Fernandes de Chantal
Andréia Campos Santana
Antônio Claudio Pimenta Ribeiro
Carolinne Januária de Souza Martins
Daniela Carina Pena Pascual
Danielle Ramos da Silva
Diogenes Ferreira Reis Fustinoni
Fábio Lúcio Lopes Mendonça
Fábio Mesquita Buiati
Glaidson Menegazzo Verzeletti
Heverson Soares de Brito
Johnatan Santos de Oliveira
Kelly Santos de Oliveira Bezerra
Luciano Pereira dos Anjos
Luciene Pereira de Cerqueira Kaipper
Luiz Antônio de Souto Evaristo
Luiz Claudio Ferreira
Marco Schaffer
Marcos Vinicius Vieira da Silva
Pedro Augusto Oliveira de Paula
Roberto Mariano de Oliveira Soares
Sergio Luiz Teixeira Camargo
Soleni Guimarães Alves
Suzane Lais De Freitas
Valério Aymoré Martins
Vera Lopes de Assis
Wladimir Rodrigues da Fonseca

HISTÓRICO DE REVISÃO

Data	Versão	Descrição
06/08/2014	0.1	Versão Inicial
16/09/2014	0.2	Revisão bibliográfica; Levantamento de padrões e normas sobre o tema;
27/10/2014	0.3	Levantamento de grupos de pesquisa sobre o tema; Levantamento de estudos de projetos na área;
10/12/2014	0.4	Estudos sobre modelos de armazenamento e acesso a informações de artefatos de identificação nacional e internacional;
04/02/2015	0.5	Estudo de protocolos de autenticação; Estudo de mecanismos de autenticação; Levantamento de Protocolos de autenticação;
31/03/2015	0.6	Estudos de identificação na Alemanha e estudo de modelo
26/05/2015	0.7	Revisão bibliográfica
01/06/2015	0.8	Versão para homologação
13/08/2015	0.9	Atualização do histórico de revisão
13/09/2015	1.0	Versão revisada

latitude
Laboratório de tecnologias da tomada de decisão

Universidade de Brasília – UnB
Campus Universitário Darcy Ribeiro - FT – ENE – Latitude
CEP 70.910-900 – Brasília-DF
Tel.: +55 61 3107-5598 – Fax: +55 61 3107-5590

SUMÁRIO

1. INTRODUÇÃO

A Secretaria Executiva (SE/MJ), vinculada ao Ministério da Justiça (MJ), é responsável por viabilizar o desenvolvimento e a implantação do Registro de Identidade Civil, instituído pela Lei nº 9.454, de 7 de abril de 1997, regulamentado pelo Decreto nº 7.166, de 5 de maio de 2010.

Atualmente, a República Federativa do Brasil conta com sistema de identificação de seus cidadãos amparado pela Lei nº 7.116, de 29 de agosto de 1983. Essa lei assegura validade nacional às Carteiras de Identidade, ou Cédulas de Identidade; confere também autonomia gerencial às Unidades Federativas no que concerne à expedição e controle dos números de registros gerais emitidos para cada documento. Essa condição de autonomia, ao contrário do que pode parecer, fragiliza o sistema de identificação, já que dá condições ao cidadão de requerer legalmente até 27 (vinte e sete) cédulas de identidades diferentes.

Com essa facilidade legal, inúmeras possibilidades fraudulentas se apresentam de maneira silenciosa, pois, na grande maioria dos casos, os Institutos de Identificação das Unidades Federativas não dispõem de protocolos e aparato tecnológico para identificar as duplicações de registro vindas de outros estados, ou até mesmo do seu próprio arquivo datiloscópico. Consoante aos fatos, os Institutos de Identificação não trabalham interativamente para que haja trocas de informações de dados e geração de conhecimento para manuseio inteligente e seguro para individualização do cidadão em prol da sociedade.

Com foco na busca de soluções para tais problemas, o Projeto RIC prevê a administração central dos dados biográficos e biométricos dos cidadãos no Cadastro Nacional de Registro de Identificação Civil (CANRIC) e ABIS (do inglês Automated Biometric Identification System), respectivamente. A previsão desse novo modelo sustenta a não duplicação de registros e a consequente identificação unívoca dos cidadãos brasileiros natos e naturalizados. O Projeto RIC, portanto, visa otimizar o sistema de identificação e individualização do cidadão brasileiro nato e naturalizado com vistas a um perfeito funcionamento da gestão de dados da sociedade, agregando valor à cidadania, à gestão administrativa, à simplificação do acesso aos serviços disponíveis ao cidadão e à segurança pública do país.

Nesse contexto, o termo de cooperação entre MJ/SE e FUB/CDT define um projeto que objetiva identificar, mapear e desenvolver parte dos processos e da infraestrutura tecnológica necessária para viabilizar a implantação do número único de Registro de

Identidade Civil – RIC no Brasil.

O presente documento tem como objetivo geral apresentar uma revisão teórica de protocolos- padrões de autenticação com criptografia, estudar os protocolos empregados em modelos de identidades eletrônicas eID (Electronic Identity) nacionais, além de analisar a segurança desses protocolos.

2. GLOSSÁRIO

AA – *Active Authentication* (Autenticação Ativa)

BAC - *Basic Access Control* (Controle de Acesso Básico)

BSI - *Bundesamts für Sicherheit in der Informationstechnik* (Escritório Federal Alemão de Segurança da Informação)

CSCA – *Country Signing Certificate Authority* (Autoridade Certificadora do País Assinante)

CVCA – *Country Verifying Certificate Authority* (Autoridade Certificadora do País Verificador)

DS – *Document Signer* (Assinante do Documento)

DES – *Data Encryption Standard* (Cifra de Blocos de Chave Simétrica)

3DES – *Triple DES* (Cifra de Blocos de Chave Simétrica que usa o DES três vezes)

EAC – *Extended Access Control* (Controle de Acesso Estendido)

e-passports – Passaportes eletrônicos

ICAO - *International Civil Aviation Organization* (Organização de Aviação Civil Internacional)

ICC – *Integrated Circuit Card* (Cartão de Cirtuito Integrado)

IFD – *Interface Device* (Dispositivo de Interface)

KA – *Key Agreement* (Função de Estabelecimento de Chaves)

KDF – *Key Derivation Function* (Função Derivadora de Chaves)

LDS – *Logical Data Structure* (Estrutura Lógica de Dados)

MAC – *Message Authentication Code* (Código de Autenticação de Mensagem)

MRTD – *Machine Readable Travel Document* (Documento de Viagem Legível por Máquina)

MRZ – *Machine Readable Zone* (Zona Legível por Máquina)

Nonce – Um número arbitrário usado uma única vez para evitar ataques de repetição

nPA -- *Neuer Personalausweis* (Novo Cartão de Identidade)

OCR – *Optical Character Recognition* (Reconhecimento Óptico de Caracteres)

PA – *Passive Authentication* (Autenticação Passiva)

PACE – *Password Authenticated Connection Establishment* (Estabelecimento de Conexão Autenticada baseado em Senhas)

PICC – *Proximity Integrated Circuit Card* (Cartão de Circuito Integrado de Proximidade)

PKD – *Public Key Directory* (Diretório de Chaves Públicas)

PKI – *Public Key Infrastructure* (Infraestrutura de Chave Pública)

SAC – *Supplemental Access Control* (Controle de Acesso Suplementar)

SO_d – *Document Security Object* (Objeto de Segurança do Documento)

3. PROTOCOLOS DE AUTENTICAÇÃO PADRONIZADOS PELA ICAO E PELO BSI

O objetivo desse capítulo é o estudo de protocolos de autenticação padronizados tanto pela Organização de Aviação Civil Internacional (ICAO – International Civil Aviation Organization) como pelo Escritório Federal Alemão de Segurança da Informação (BSI – Bundesamts für Sicherheit in der Informationstechnik), e mundialmente utilizados em passaportes eletrônicos (e-passports). Tal estudo é motivado pelo amplo uso desses protocolos em diversos cartões eletrônicos de identidade (cartões eID) da Europa.

Existem três gerações de passaportes eletrônicos. Os passaportes da primeira geração, lançados em 2005, dispõem de três tipos de autenticação: autenticação passiva, autenticação ativa, e um mecanismo de segurança chamado Controle de Acesso Básico (BAC – Basic Access Control). Esses protocolos são definidos no DOC 9303 da ICAO [2]. Já os passaportes da segunda geração, lançados em 2009, utilizam um mecanismo de segurança extra, chamado de Controle de Acesso Estendido (EAC – Extended Access Control) [4,5]. Tal mecanismo foi introduzido pelo BSI. Os passaportes de terceira geração deverão ser lançados, no final de 2014, e contarão com um forte mecanismo de segurança, chamado de Controle de Acesso Suplementar (SAC – Supplemental Access Control) [3]. O SAC foi definido pela ICAO em 2010 para substituir o BAC; ele é baseado num protocolo de estabelecimento de conexão baseado em senhas, o PACE (Password-based Authenticated Connection Establishment).

Infraestruturas de Chave Pública -- A fim de facilitar o processo de autenticação dos passaportes eletrônicos de primeira, segunda e terceira geração, duas infraestruturas de chave pública (PKI – Public Key Infrastructure) são necessárias: a PKI da CSCA (Country Signing Certificate Authority – Autoridade Certificadora do País Assinante) e a PKI da CVCA (Country Verifying Certificate Authority – Autoridade Certificadora do País Verificador). Sucintamente, uma PKI é uma hierarquia de certificados digitais, ou seja, existe uma Autoridade Certificadora raiz (CA – Certificate Authority) que emite um certificado raiz para si própria e certificados para autoridades um nível abaixo da CA raiz, as quais, por conseguinte, emitem certificados para autoridades um nível abaixo. A cadeia de certificados de uma PKI pode ser arbitrariamente longa e tem a propriedade de que o certificado raiz pode ser usado como uma âncora de confiança, já que a Autoridade Certificadora raiz é

digna de confiança de todos os participantes da PKI. Os princípios de uma PKI são baseados em criptografia assimétrica, na qual cada participante da PKI tem um par de chaves pública e privada e pode usar a chave privada para a emissão (através de assinatura digital) de certificados digitais. O certificado digital contém: a chave pública do participante; informação de quem emitiu o certificado (quem assinou o certificado) e para quem foi emitido (o dono da chave privada correspondente à chave pública do participante); o período de validade, dentre outras informações, além da assinatura desses dados pela CA imediatamente superior. A PKI da CSCA é utilizada para verificar se os dados do chip de um e-passport foram produzidos por ordem oficial do país emissor ou não, e se os dados foram alterados ou não. Para que sejam válidos, tais dados tem que ter sido digitalmente assinados por uma autoridade certificadora na PKI da CSCA. Essa PKI é usada no protocolo de **Autenticação Passiva**. A Figura 3.1 mostra a PKI da CSCA.

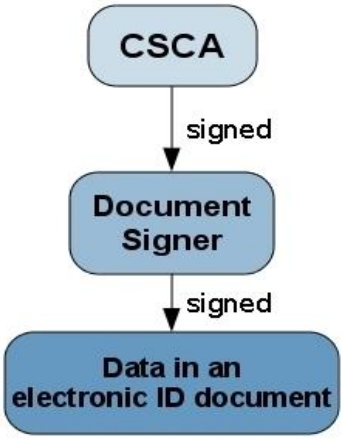

Figura 3.1 (tirada da página do BSI [7]): PKI da CSCA.

Na PKI da CSCA, a CSCA auto-assina o certificado raiz, C_CSCA, contendo sua chave pública, e também emite Certificados para Assinantes de Documentos (DS – Document Signer), os C_DS. Os DSs são os emissores de passaportes. Esses por sua vez usam suas chaves privadas para assinar dados do documento eletrônico. Para que se possa provar a autenticidade e integridade de passaportes em controles de fronteira, os países tem que trocar entre si seus certificados raízes de maneira segura. Isso é feito através de intercâmbio diplomático ou através de uma plataforma oficial da ICAO, a ICAO-PKD (ICAO Public Key Directory – Diretório de Chaves Públicas da ICAO), que pode ser acessada por países participantes. Cada país publica nessa PKD seu C_CSCA, os C_DS

e uma lista de certificados revogados (CRL – Certificate Revocation List). Vale salientar que na Alemanha a CSCA é o BSI.

Já a CVCA é uma hierarquia de certificados que serve para verificar se uma certa autoridade tem permissão de acessar dados sensíveis do chip. A PKI da CVCA é usada no mecanismo de Controle de Acesso Estendido, **EAC**. A Figura 3.2 mostra a PKI da CVCA.

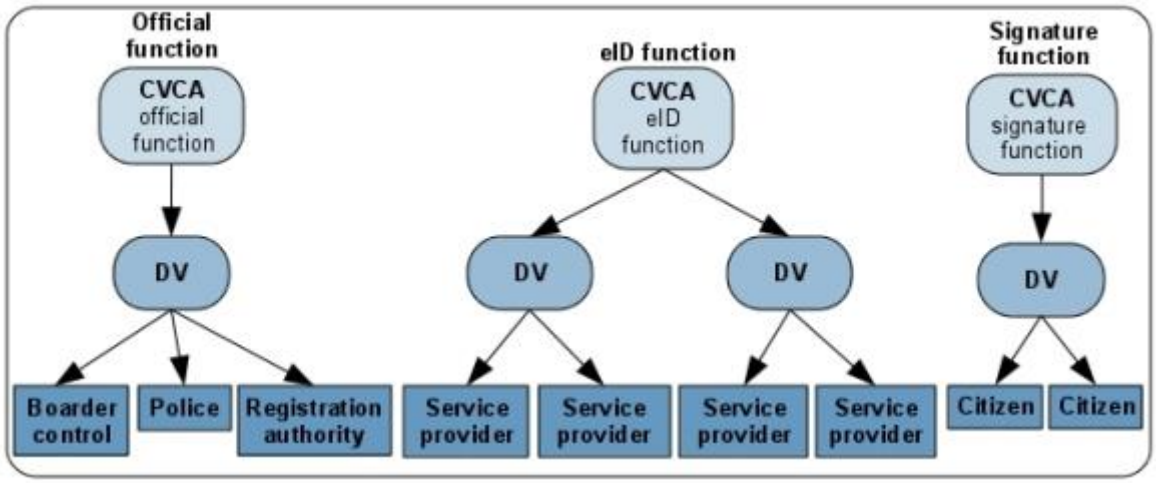

Figura 3.2 (tirada da página do BSI [7]): PKI da CVCA.

A CVCA também é operada pelo BSI (vale enfatizar que mesmo que a CSCA e a CVCA sejam integradas numa entidade única, elas devem usar pares distintos de chave pública e privada). Sua chave privada é usada para auto-assinar o certificado raiz C_CVCA, e também para assinar Certificados dos Verificadores de Documentos (DV – Document Verifier), os C_DV. Os DV são autoridades certificadoras autorizadas pela CVCA para emitir certificados de autorização de leitura de documentos eletrônicos para terminais ou sistemas de inspeção. Tais certificados, assim como os C_DV, também especificam tipos de permissão de leitura, ou seja, que dados o DV ou terminal tem permissão de acessar. Os certificados de autorização podem ser emitidos para funções oficiais, como também para funções de eID, como é o caso em serviços de governo eletrônico e comércio eletrônico, e também para funções de assinatura digital. No caso de funções oficiais, o CVCA de uma país A também pode emitir certificados para DVs de um país B. Isso é feito para que o país B tenha permissão de acesso a dados sensíveis de chips de passaportes eletrônicos emitidos pelo país A. Para validar um certificado de autorização de um determinado terminal, o chip do passaporte eletrônico deve verificar uma cadeia de certificados, e para isso deve conter um certificado de verificação de confiança. Por exemplo, se o chip contém

o C_CVCA (armazenado em sua memória na fase de pré-personalização do documento), e recebe uma cadeia de certificados de um terminal de controle de fronteira, o chip pode facilmente verificar que esse último terminal é autorizado à leitura de dados sensíveis do chip. Suponha que a cadeia de certificados seja composta por C_T (certificado do terminal), C_DV, e C_CVCA. Para verificação o chip executa os seguintes passos: 1. Lê o C_T e verifica qual o DV que assinou o C_T; 2. Lê o C_DV, que foi assinado pela CVCA, para encontrar a chave pública do DV; 3. Verifica que o C_T realmente foi emitido pelo DV; 4. Lê o C_CVCA para encontrar a chave pública da CVCA; 5. Verifica que o C_DV foi realmente emitido pela CVCA.

A seguir descreveremos cada um dos mecanismos de autenticação citados acima. O conteúdo das Seções 3.1, 3.2 e 3.3 se baseia nos documentos da ICAO DOC 9303 Parte 1, Versões 1 e 2 [1,2], enquanto que o conteúdo da Seção 3.4 se baseia nos relatórios técnicos do BSI TR-03110-1 e TR-03110-2 [4,5]. Seção 3.5 se baseia no relatório técnico da ICAO TR-SAC 1.1 [3].

3.1. Autenticação Passiva (Obrigatória pela ICAO)

Antes de começarmos a explicar o que é e como funciona a autenticação passiva, é imprescindível esclarecermos alguns dos conteúdos armazenados no chip de um e-passport. Para assegurar que chips programados em países diferentes possam ser lidos em qualquer outro país, uma estrutura lógica de dados (LDS – Logical Data Structure) foi especificada pela ICAO. A LDS descreve como dados são escritos e formatados nos e-passports. Ela define uma série de Grupos de Dados (DGs – Data Groups), obrigatórios e opcionais (DG1 a DG16). A estrutura lógica de dados, LDS, pode ser vista na Figura 3.3.

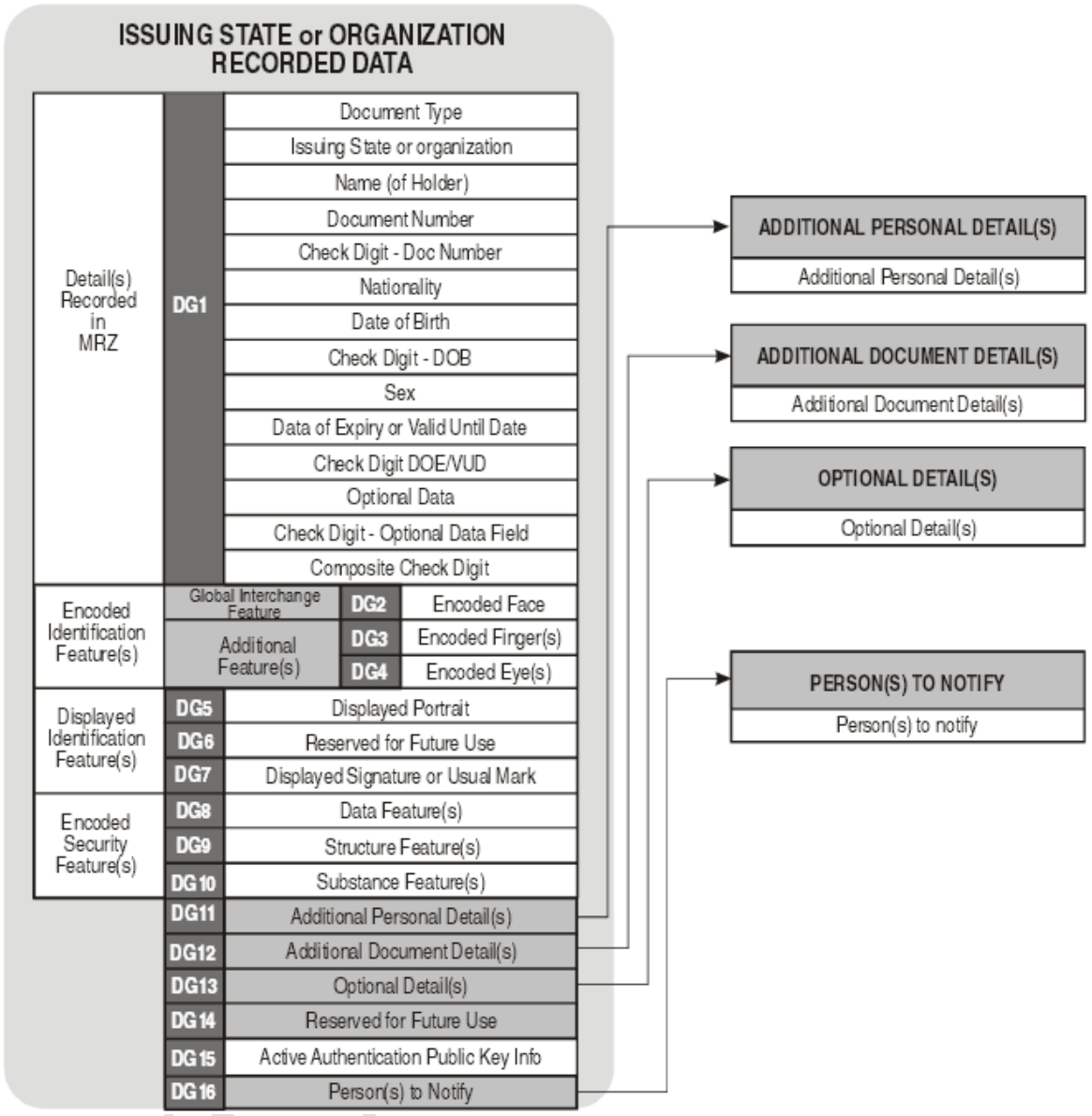

Figura 3.3 (tirada do DOC 9303 da ICAO [2]): LDS.

Além dos Grupos de Dados da LDS, o chip também armazena um Objeto de Segurança do Documento, SO_d (Document Security Object), o qual contém um hash de cada Grupo de Dados da LDS digitalmente assinado pelo país emissor do passaporte. Efetivamente, esse grupo de dados é assinado por uma autoridade emissora de passaporte, ou seja, um DS integrante da PKI da CSCA. O nome do DS e a assinatura digital calculada por ele são outros dos vários elementos contidos no SO_d.

A autenticação passiva (ou PA, Passive Authentication) é um mecanismo criptográfico utilizado para provar que os conteúdos do SO_d e da LDS são autênticos e

não foram alterados, ou seja, esse tipo de autenticação prova que o conteúdo do chip é autêntico. Para compreendermos como a PA funciona, e com o objetivo de facilitar a leitura de referência bibliográfica, explicaremos com mais detalhes o protocolo utilizando a mesma notação usada no documento da ICAO DOC 9303 Parte 1, Volume 2 [2].

O protocolo de Autenticação Passiva funciona da seguinte forma.

1. Primeiramente, o Sistema de Inspeção, ou terminal, quer se assegurar que o SO_d contido no chip é autêntico.
2. Para isso o Sistema de Inspeção (IS – Inspection System) lê o Objeto de Segurança do Documento, SO_d, do chip. O SO_d contém o nome do assinante do documento (DS – Document Signer) e opcionalmente o Certificado do Assinante do Documento (C_DS).
3. O Sistema de Inspeção utiliza a Chave Pública do Assinante do Documento (KPu_DS), contida no C_DS, para verificar a assinatura digital do Objeto de Segurança do Documento, SO_d. (O certificado do assinante do documento da chave KPu_DS é armazenado no Sistema de Inspeção, baixado do Diretório de Chaves Públicas (PKD - Public Key Directory) da ICAO. Esse certificado também pode ser armazenado no chip do MRTD (Machine Readable Travel Document – Documento de Viagem Legível por Máquina).) Isso assegura que o SO_d é autêntico, foi realmente emitido pela autoridade mencionada nele, e não foi alterado.
4. Agora o Sistema de Inspeção precisa verificar se os conteúdos dos Grupos de Dados da LDS são autênticos.
5. Para isso o Sistema de Inspeção lê os Grupos de Dados relevantes da LDS e calcula os hashes dos conteúdos.
6. O IS compara o resultado com os hashes correspondentes armazenados no SO_d. Caso os valores sejam iguais, o IS se assegura que os conteúdos dos Grupos de Dados da LDS são autênticos e não foram alterados.

Esse tipo de mecanismo é chamado de Autenticação Passiva pois não requer que o chip do MRTD tenha capacidade de processamento. Em algumas referências, ele também

é chamado de Autenticação Assimétrica Estática.

Note que para que seja possível executar o protocolo PA, o Sistema de Inspeção precisa ter acesso às Chaves Públicas dos países emissores. Isso é necessário para que o IS possa verificar a autenticidade dos Certificados dos Assinantes de Documentos (C_DS). Portanto, o Certificado da Autoridade Certificadora de cada País Assinante (Country Signing Certificate Authority) (C_CSCA), juntamente com os Certificados dos Assinantes de Documentos (C_DS), devem ser armazenados no Sistema de Inspeção.
Uma deficiência da Autenticação Passiva é que ela não impede cópia e/ou substituição do chip.

3.2. Autenticação Ativa (Opcional pela ICAO)

Para entendermos o protocolo de Autenticação Ativa, precisamos conhecer um outro conteúdo do chip do MRTD, os dados contidos na MRZ (Machine Readable Zone – Zona Legível por Máquina). Todo MRTD contém um campo de dados, obrigatórios e opcionais, para leitura por máquinas que utilizam métodos de Reconhecimento Óptico de Caracteres (OCR – Optical Character Recognition). Os dados do MRZ são armazenados no chip no Grupo de Dados 1 (DG1) da LDS, e são também impressos na página de dados do passaporte. Os dados do MRZ podem ser lidos visualmente e também são formatados num formato padrão de forma que possam ser lidos por leitoras internacionalmente, desde que as leitoras sejam configuradas de acordo com o Documento 9303 da ICAO. A Figura 3.4 mostra um exemplo de uma página de dados de um passaporte destacando o campo MRZ, e a Figura 3.5 mostra em detalhes cada elemento do MRZ.

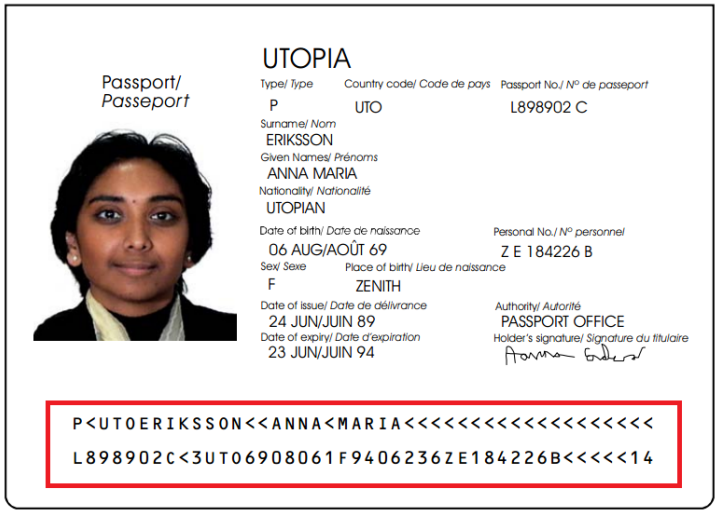

Figura 3.4 (tirada do DOC 9303 da ICAO [1]): Exemplo de página de dados de um passaporte destacando o MRZ.

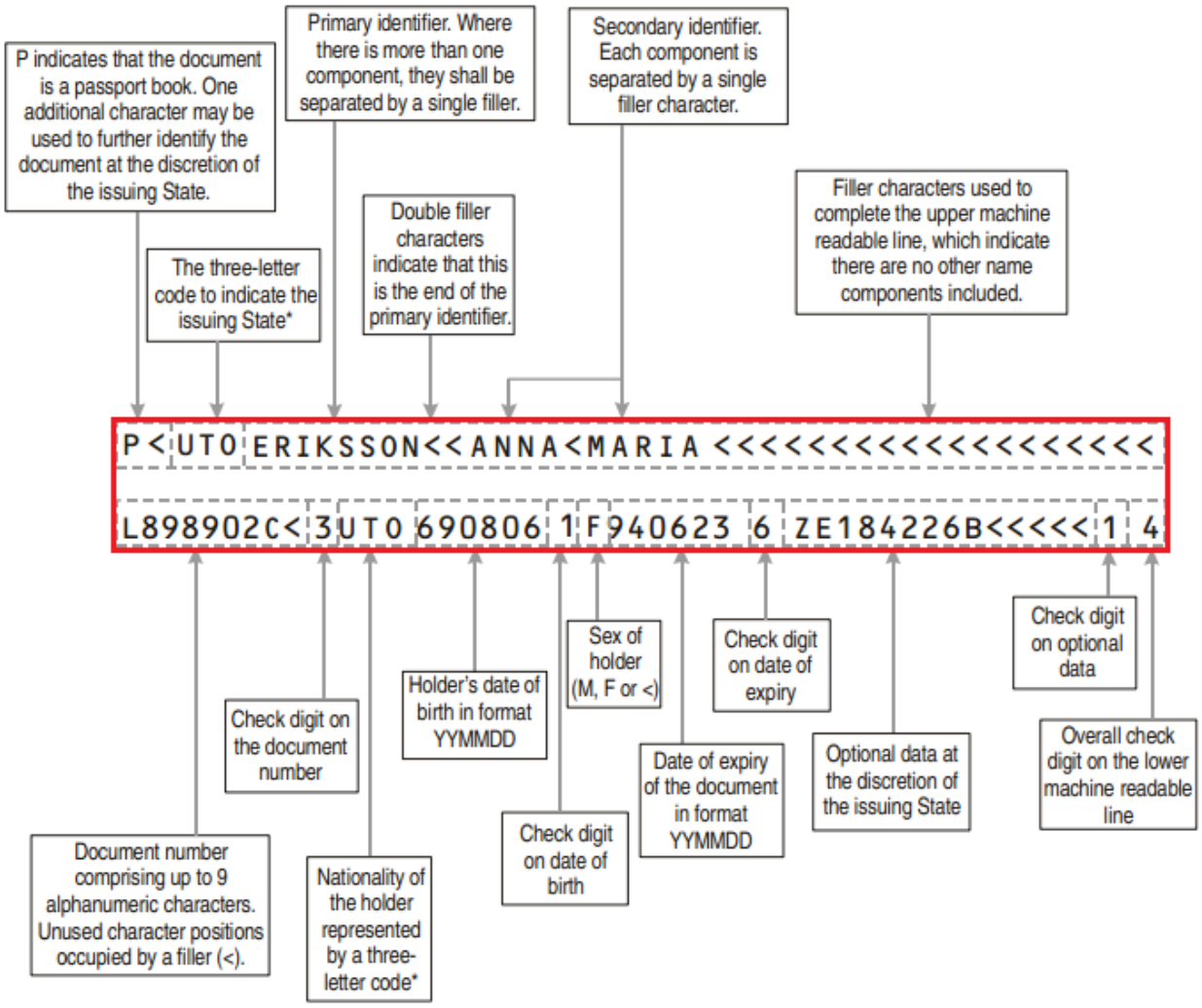

Figura 3.5 (tirada do DOC 9303 da ICAO [1]): Estrutura do MRZ.

O protocolo de Autenticação Ativa (AA – Active Authentication) é um protocolo de desafio-resposta entre o chip do MRTD e o Sistema de Inspeção, que tem como finalidade assegurar que o chip é genuíno e não tenha sido substituído. Esse protocolo verifica que os dados eletrônicos contidos pertencem ao documento físico e ao chip físico. Para isso, o chip que suporta esse tipo de protocolo contém seu próprio par de chaves privada e pública para Autenticação Ativa, KPr_AA e KPu_AA (Recomenda-se que esse par de chaves seja usado somente para a Autenticação Ativa. Como veremos a seguir, o protocolo AA permite que o chip assine qualquer mensagem que o terminal queira, inclusive informação semântica que permite a rastreabilidade do mesmo). A chave pública KPu_AA é armazenada no Grupo de Dados 15 (DG15) da LDS, e a chave privada KPr_AA numa memória segura do chip.

O protocolo de Autenticação Ativa funciona da seguinte forma.

1. Primeiramente, o Sistema de Inspeção tem como objetivo assegurar que os dados do MRZ da página de dados são os mesmos que os contidos no chip, ou seja, que o chip não foi substituído.
2. Para isso, o IS lê o MRZ inteiro da página de dados do MRTD (se o ainda não tiver feito) e compara com o valor do MRZ armazenado no chip no Grupo de Dados 1 (DG1). Vale lembrar que a autenticidade e integridade do DG1 já foi comprovada pela Autenticação Passiva.
3. Agora, o Sistema de Inspeção precisa provar que o SO_d não é uma cópia, e o chip é genuíno.
4. Para isso, o IS executa um protocolo de desafio-resposta entre ele e o chip. Esse protocolo faz uso do par de chaves de Autenticação Ativa do chip (KPr_AA e KPu_AA). A chave pública Pku_AA é acessível ao IS, e sua autenticidade e integridade já foi comprovada pela Autenticação Passiva, pois PKu_AA está armazenada no DG15. No entanto a chave privada não pode ser lida. Só o MRTD original tem conhecimento dessa chave, pois ela é armazenada numa memória segura do chip. Uma versão simplificada do protocolo de desafio-resposta é ilustrada na Figura 3.6.

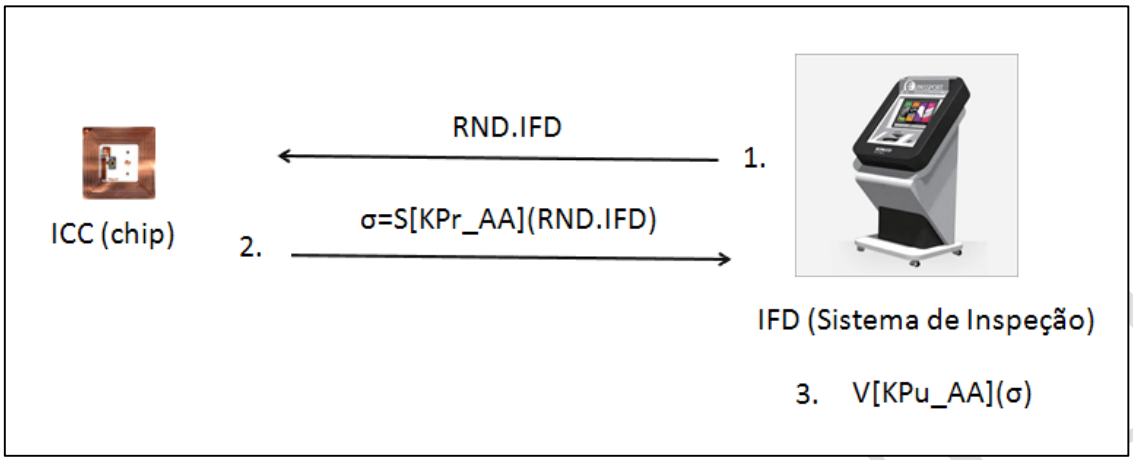

Figura 3.6: Protocolo de autenticação ativa simplificado.

Na Figura 3.6 vemos que o Sistema de Inspeção (o Dispositivo de Interface, IFD – Interface Device) envia um desafio RND.IFD para o chip (o Integrated Circuit Card, ICC – Cartão de Circuito Integrado). Esse desafio deve ser um nonce de 8 bytes. O chip usa sua Chave Privada de Autenticação Ativa, KPr_AA, para calcular a assinatura digital do nonce RND.IFD e envia a assinatura para o Sistema de Inspeção. O IS então usa a Chave Pública de Autenticação Ativa do chip, KPu_AA, para verificar se a assinatura é válida. Só o chip genuíno poderia ter assinado o desafio, já que a chave privada KPr_AA fica armazenada numa parte segura do chip.

De uma forma mais completa, o protocolo de desafio-resposta entre o chip e o IS funciona da seguinte forma.

1. O IS gera um nonce de 8 bytes RND.IFD e manda para o chip.
2. O chip, ao invés de assinar apenas RNF.IFD, assina uma mensagem representativa F, por razões de segurança. Essa mensagem é composta de um cabeçalho (header), um nonce RND.ICC, o hash de RND.ICC concatenado com RND.IFD, H, e um reboque T (trailer). O método de assinatura é especificado pelo padrão ISO/IEC 9796-2. Se o algoritmo SHA-1 for usado como função hash, o padrão ISO/IEC 9796-2 especifica (para um mecanismo de assinatura baseado em fatoração de inteiros, como o RSA) o header='6A', e o trailer T='BC'. A mensagem representativa F seria, portanto:

F = ‘6A’ || RND.ICC || SHA-1(RND.ICC || RND.IFD) || ’BC’

No caso do algoritmo RSA ser usado, o tamanho de F deve ser igual ao tamanho do módulo RSA, k (em bytes), e o tamanho do nonce RND.ICC é, portanto, L_RND.ICC = k - L_H - 2 bytes, onde L_H é o comprimento de saída da função hash. Note que para um determinado algoritmo de assinatura digital, deve-se escolher uma função hash com tamanho de saída apropriado. Por exemplo, RSA 1024 com SHA-1, são compatíveis já que 1024 (bits) > 160 + 16 (bits).

3. O IFD então decifra a assinatura com a chave pública do chip, extrai RND.ICC e H, calcula o hash de RND.ICC || RND.IFD e verifica se ele é igual ao hash H.

Note que para que seja possível executar o protocolo AA, todas as informações necessárias já estão contidas no MRTD, e, portanto, o Sistema de Inspeção não precisa de informação externa adicional. No entanto, o protocolo AA requer que o chip tenha capacidade de processamento para que se possa assinar o nonce enviado pelo Sistema de Inspeção. Esse mecanismo de autenticação também é chamado em algumas referências de Autenticação Assimétrica Dinâmica.

3.3. Controle de Acesso Básico (BAC)

O protocolo BAC (Basic Access Control) é um mecanismo de controle de acesso que impede que o chip sem contato do MRTD seja lido sem que o portador do documento tenha dado seu consentimento (entregando o MRTD ao Sistema de Inspeção), e também impede que a comunicação entre o chip do MRTD e a máquina leitora seja interceptada. Em outras palavras, o BAC impede ataques de skimming e eavesdropping.

Skimming é um tipo de ataque em que o atacante adquire dados do chip sem que esteja em posse do MRTD e sem que tenha permissão do portador do MRTD. Ele é considerado como sendo um ataque online, já que o atacante precisa se comunicar com o chip durante o ataque. No caso dos ataques de Eavesdropping (ou ataques de interceptação), dados são interceptados pelo atacante enquanto o chip se comunica com a leitora. Esse é considerado como sendo um ataque offline, já que os dados só são analisados depois da interceptação da comunicação.

O BAC é um protocolo de autenticação mútua e que proporciona troca segura de mensagens. Ele funciona da seguinte forma.

4. Primeiro, a partir de informações contidas na Zona de Leitura por Máquina (MRZ), uma chave-simétrica é gerada para autenticação mútua. Essa chave, chamada de chave semente, ou K_seed, é usada para determinar duas outras chaves: K_ENC e K_MAC. A chave K_ENC é usada para cifragem de mensagens usando 3DES e a chave K_MAC é usada para cálculo do valor MAC (Message Authentication Code – Código de Autenticação de Mensagens) de acordo com ISO/IEC9797-1.

5. O Sistema de Inspeção prova que é autorizado a acessar o chip através de um protocolo desafio-resposta. Isso mostra que o IS tem acesso as chaves individuais do chip de Acesso Básico ao Documento (K_ENC e K_MAC), derivadas do MRZ. A informação do MRZ tem que ter sido previamente obtida através de leitura óptica ou visual do campo MRZ do MRTD.

6. Depois que o Sistema de Inspeção tenha sido autenticado com sucesso, o chip também se autentica no mesmo protocolo desafio-resposta e calcula chaves de sessão (KS_ENC e KS_MAC) a partir de informações obtidas da autenticação do Sistema de Inspeção. A geração de chaves de sessão também faz parte do protocolo desafio-resposta entre o chip e o Sistema de Inspeção.

7. O Sistema de Inspeção usa informações obtidas da autenticação do chip para calcular as chaves de sessão (KS_ENC e KS_MAC).

8. Depois da autenticação mútua e do estabelecimento de chaves de sessão, o Sistema de Inspeção lê os dados do chip através de um canal seguro (usando um mecanismo chamado **Secure Messaging**).

Descreveremos agora passo a passo cada um dos procedimentos do mecanismo BAC.

Criação da Chave Semente K_seed inicial: para a autenticação do Sistema de Inspeção, a chave semente K_seed é gerada a partir de informações contidas no MRZ. Primeiramente, o Sistema de Inspeção lê a “Informação do MRZ” consistindo do Número do Documento, Data de Nascimento e Data de Expiração, incluindo os seus dígitos de

verificação (Informações detalhadas da estrutura do MRZ podem ser encontradas no documento “DOC 9303 - Parte 1, Volume 1” da ICAO [1]). Para isso, o IS usa leitura óptica ou a informação pode ser lida visualmente e digitada. O IS então calcula o hash SHA-1 da “Informação do MRZ” e usa os 16 bytes mais significativos do resultado como sendo a chave semente K_seed, isto é, **K_seed = SHA-1[16bytes](“Informação do MRZ**”). A Figura 3.7 mostra a “Informação do MRZ” necessária para o cálculo de K_seed.

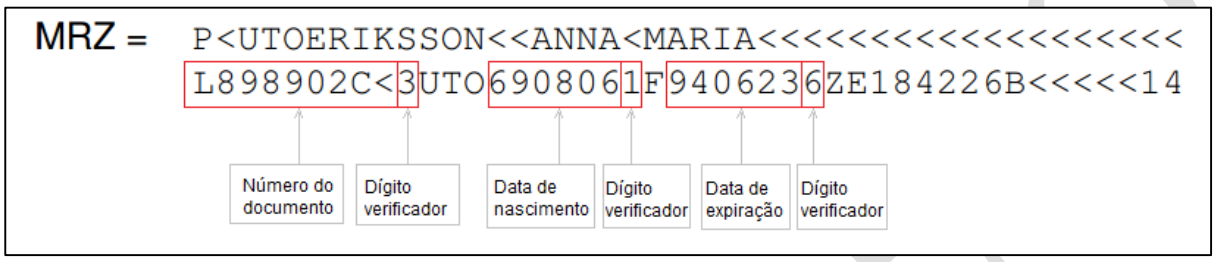

Figura 3.7 (tirada do DOC 9303 da ICAO [1]): “Informação do MRZ”.

Geração das Chaves de Autenticação Básica K_ENC e K_MAC: o cálculo das chaves de Acesso Básico ao Documento, K_ENC e K_BAC é feito a partir da chave semente K_seed e de um contador de 32 bits, c. O valor c = 0x 00 00 00 01, em hexadecimal, é usado para o cálculo de K_ENC, enquanto o valor c = 0x 00 00 00 02 é usado para o cálculo de K_MAC. A chave K (que pode ser tanto K_ENC quanto K_MAC, dependendo do valor de c) é obtida da seguinte forma: Primeiro calcula-se o hash SHA-1[16bytes](K_seed || c), isto é, os 16 bytes mais significantes do hash SHA-1 da concatenação de K_seed com c. Depois divide-se esse valor em dois valores de 8 bytes cada, K_a e K_b. Para finalizar, ajusta-se os bits de paridade de K_a e K_b de forma a formar chaves corretas tipo DES. As chaves do DES usam 8 bytes, nos quais 7 bits de cada byte são usados para a chave. O bit menos significativo é usado para paridade e é jogado fora para a cifragem/decifragem, resultando em uma chave de 56 bits para o DES. O cálculo de paridade do DES funciona da seguinte forma: para cada byte conte o número de bits que são “1” e mude o bit menos significativo se a contagem der um número ímpar. Note que cada chave K_ENC e K_MAC é a concatenação de K_a e K_b, formando então 128 bits cada (16 bytes). (A chave K_ENC é dividida em duas chaves de 8 bytes K_a e K_b porque essas últimas serão usadas como chaves DES no criptosistema 3DES.) As chaves de sessão para troca de mensagens usando um canal seguro, KS_ENC e KS_MAC, são geradas da mesma forma que as chaves de Acesso Básico, K_ENC e K_MAC. A Figura 3.8

ilustra a formação de chaves K_ENC e K_MAC, e/ou KS_ENC e KS_MAC.

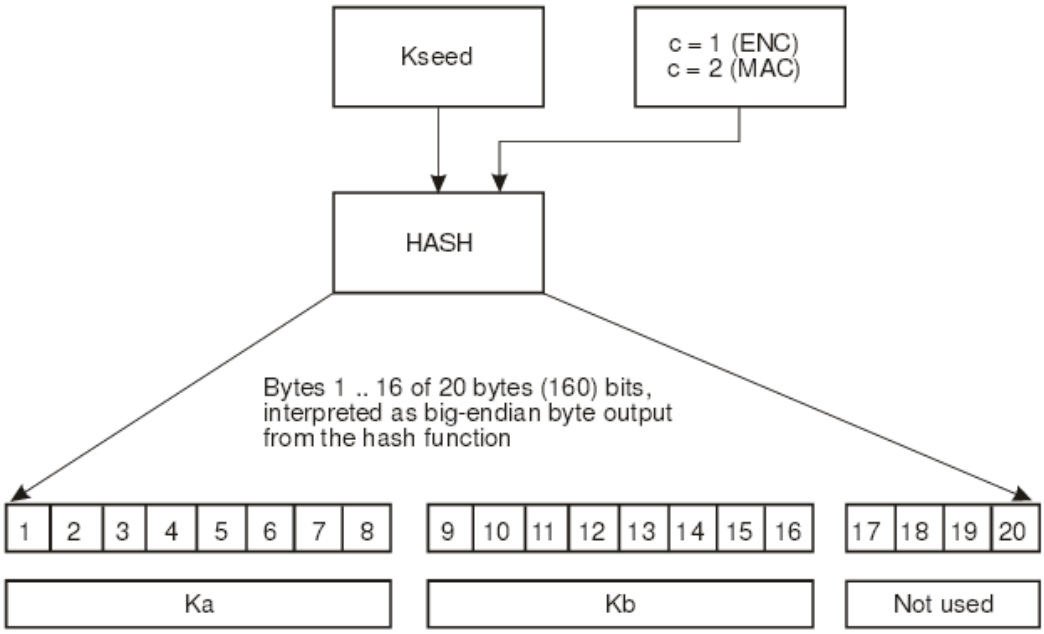

Figura 3.8 (tirada do DOC 9303 da ICAO [2]): Formação de Chaves de Acesso Básico.

Protocolo de desafio-resposta para autenticação mútua e obtenção das chaves de sessão: ele é um protocolo desafio-resposta de três passos (pois só troca três mensagens de dados) e usa o 3DES como cifra de bloco e um algoritmo MAC como soma de verificação. A Figura 3.9 ilustra o protocolo entre o IFD (Interface Device – Dispositivo de Interface), ou Sistema de Inspeção, e o ICC (Integrated Circuit Card – Cartão de Circuito Integrado), ou chip.

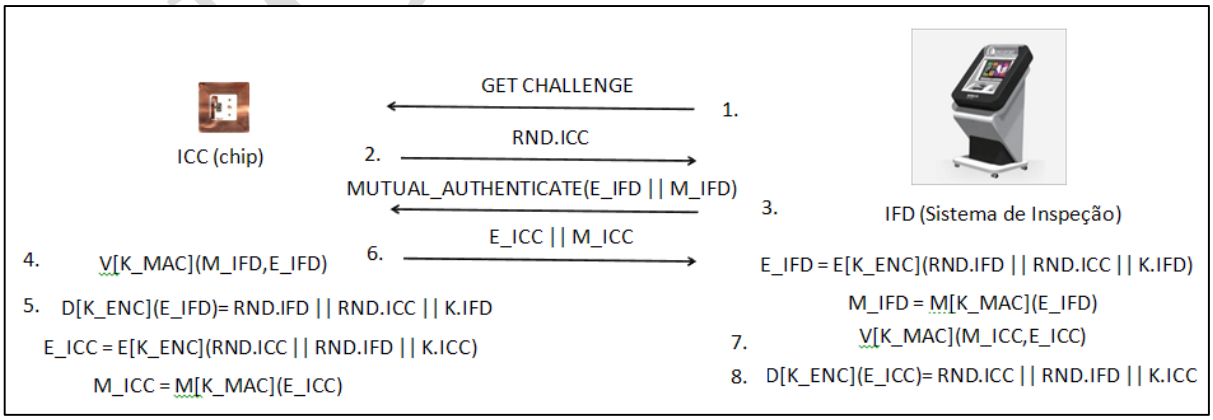

Figura 3.9: Protocolo desafio-resposta do BAC.

Na Figura 3.9 vemos que o Sistema de Inspeção solicita que o chip envie um desafio RND.ICC, que é um nonce de 8 bytes. Após receber o nonce do chip, o Sistema de Inspeção

gera um nonce RND.IFD, também de 8 bytes, e um material de chave K.IFD de 16 bytes. O IS (ou IFD) usa a chave de Acesso Básico ao Documento K_ENC para cifrar a concatenação dos valores RND.IFD, RND.ICC e K.IFD (um total de 32 bytes, ou 256 bits) com a cifra de blocos 3DES, e depois calcula o MAC do resultado usando K_MAC. A concatenação do texto cifrado e do MAC são enviados ao chip (ou ICC). O chip então verifica o MAC, usando K_MAC, decifra o texto-cifrado e verifica se o nonce RND.ICC é o mesmo que o enviado por ele. Essa parte do protocolo assegura a autenticação do Sistema de Inspeção, já que ele tem acesso as chaves K_ENC e K_MAC para cifrar o desafio RND.ICC e posteriormente calcular o MAC do resultado. Após a autenticação do Sistema de Inspeção, é a vez do chip se autenticar. Para isso ele segue procedimentos similares que o IFD. No final do protocolo, ambos o chip e o Sistema de Inspeção obtêm uma nova chave semente K_seed' = K_ICC xor K_IFD que será usada para obter as chaves de sessão KS_ENC e KS_MAC usadas para a leitura do chip de forma segura.

Uma vantagem do protocolo BAC é que ele não requer uma Infraestrutura de Chaves Públicas (PKI – Public Key Infrastructure). No entanto, o BAC tem algumas limitações. Uma delas é que ele não detecta a clonagem de chips. Outra é que a entropia da fonte de geração da chave semente (MRZ) é baixa (em geral menos de 73 bits e em muitos casos muito abaixo de 56 bits) e estática, e por isso pode ser vulnerável a ataques de força bruta.

Ataque de força bruta contra o BAC: a comunicação de rádio frequência (RFID) entre o chip e o terminal é feita através de dois canais: o canal da leitora para o e-passport, que fornece energia para o e-passport e é usado para transferência de dados da leitora para o e-passport, e o canal do e-passport para a leitora, que é usado pelo e-passport para enviar dados para a leitora. Embora o sinal no canal de comunicação do e-passport para é leitora seja mais fraco que o sinal no canal de comunicação da leitora para o e-passport, a interceptação dos dois canais de uma distância de vários metros (pelo menos 4 metros) é uma ameaça real. A arquitetura do ataque de força bruta contra o BAC pode ser dividida em duas partes: por um lado temos um interceptador que lê e grava a comunicação entre o e-passport e a leitora, e por outro lado temos um sistema de criptoanálise que recebe pares de mensagens e cifragens das mesmas e tenta obter as chaves do BAC. Numa abordagem baseada na interceptação da comunicação bilateral entre o chip e a leitora, o interceptador primeiramente obtém as mensagens: RND.ICC, E_IFD||M_IFD,

E_ICC||M_ICC e toda a comunicação cifrada pela chave K_ENC (ver Figura 3.9). Depois o sistema de criptoanálise, tendo recebido essas mensagens, faz uma busca da "Informação do MRZ" para achar um valor cuja cifragem corresponda aos 8 bytes mais significativos de E_ICC. Mais especificamente, ele calcula E*=E[K](RND.ICC), em que K denota os possíveis candidatos para a chave K_ENC (derivada da "Informação do MRZ"), e denota a cifragem com 3DES. Se msB_8(E_ICC)=E*, então K=K_ENC, e, portanto, toda a comunicação cifrada entre o chip e o terminal pode ser decifrada, obtendo-se assim os dados do e-passport. Para entendermos a razão da verificação msB_8(E_ICC)= E[K](RND.ICC), note que E_ICC=E[K_ENC](RND_ICC||RND_IFD||K.ICC), onde E denota 3DES no modo CBC (Cipher Block Chaining) com IV=0 (ver Figura 3.10). Como no 3DES a cifragem é feita em blocos de 8 bytes, se usarmos o modo CBC com IV=0, os 8 bytes mais significativos de E_ICC serão iguais ao resultado da cifragem de RND_ICC (também com 8 bytes) com a chave K_ENC.

Uma segunda abordagem de ataque de força bruta é baseada na interceptação somente do canal de comunicação da leitora para o e-passport. Essa abordagem é usada quando a distância do interceptador ultrapassa o limite da distância de interceptação do canal do e-passport para a leitora. O interceptador obtém E_IFD||M_IFD e o sistema de criptoanálise verifica por força bruta se MAC[K](E_IFD)=M_IFD, em que K é uma chave candidata a K_MAC. Apesar de nesse tipo de ataque o interceptador não conseguir obter os dados do e-passport, ele consegue fazer o rastreamento do documento através de K_MAC.

Dependendo do país emissor do e-passport, a entropia da "Informação do MRZ" pode ser tão baixa quanto 35 bits. Em [9] os autores estimam que em 2019 (quando um e-passport com período de validade de 5 anos irá expirar, assumindo que ele é emitido hoje) chaves com entropia de 35 bits poderão ser quebradas em alguns segundos com quase nenhum custo (aproximadamente 100 dólares). Maiores informações sobre ataques de força bruta contra o BAC podem ser encontradas em [8].

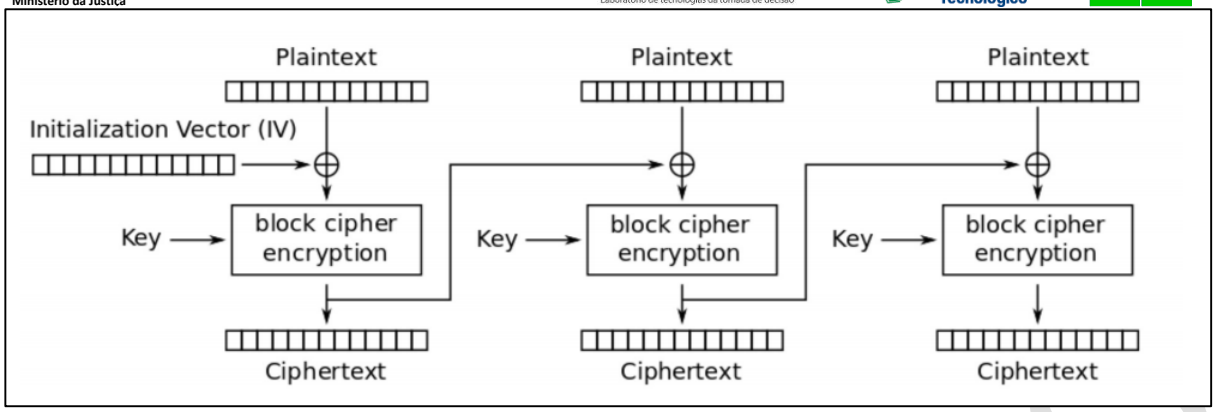

Figura 3.10. Modo de cifragem CBC.

3.4. Controle de Acesso Estendido (EAC)

O EAC é um mecanismo de autenticação adicional para fortalecer a proteção da privacidade de dados sensíveis contidos no chip do MRTD, como imagens de impressões digitais ou imagens de íris. O EAC garante que esse tipo do dado só pode ser acessado por terminais autorizados. Essencialmente, a diferença entre o controle de acesso do BAC e o do EAC é que o BAC verifica se o terminal tem acesso físico ao documento (através da leitura óptica do MRZ), enquanto que o EAC verifica se o terminal tem permissão de acesso a dados sensíveis do chip. O EAC é baseado em criptografia de chave pública e requer uma infraestrutura de chave pública para as leitoras de chip. Ele consiste de dois protocolos: um protocolo para autenticação do chip (CA – Chip Authentication) e um protocolo para autenticação do terminal (TA – Terminal Authentication).

Protocolo de autenticação do chip: protocolo de estabelecimento de chaves Diffie-Hellman usado para estabelecer uma conexão segura entre o chip e a leitora, e para fazer autenticação unilateral do chip, detectando chips clonados. Para que a execução desse protocolo seja possível, o chip deve conter um par de chaves pública e privada especial, PK_PICC e SK_PICC. A chave privada é armazenada numa área segura do chip que não pode ser lida; mesmo se o chip inteiro for clonado (copiado), não é possível copiar SK_PICC.

Protocolo de autenticação do terminal: protocolo desafio-resposta de dois passos usado para autenticação unilateral explícita do terminal. Esse protocolo permite que o chip

verifique se o terminal tem permissão de acessar certos dados sensíveis. Para isso, o terminal prova que tem a chave privada, SK_IFD, associada a determinada chave pública, PK_IFD, através da assinatura digital de um desafio enviado pelo chip. Para que o chip possa verificar se o terminal dono da chave pública PK_IFD tem realmente permissão de acesso, o chip deve ter armazenado em sua memória o certificado raiz da PKI da CVCA (uma hierarquia de certificados de autorização para leitura de dados sensíveis), ou seja, o C_CVCA.

O BSI padroniza duas versões do EAC: EACv1 e EACv2. A seguir descreveremos em detalhes os protocolos das versões 1 e 2 do EAC.

3.4.1. Autenticação do Chip Versão 1 (CAv1)

No mecanismo EACv1, a autenticação do chip (CAv1) é feita antes da autenticação do terminal (TAv1). O protocolo CAv1 funciona da seguinte forma.

- O chip envia para o terminal a sua chave pública (estática) PK_PICC e uma conjunto de parâmetros para estabelecimento de chaves Diffie-Hellman, D_PICC.
- O terminal usa D_ PICC para gerar um par de chaves (efêmeras) do tipo Diffie-Hellman, PKe_IFD, SKe_IFD, e envia PKe_IFD para o chip.
- Ambos, o chip e o terminal, calculam o seguinte:

a) a chave simétrica secreta K = KA(SK_PICC, PKe_IFD, D_PICC) = KA(SKe_IFD, PK_PICC, D_PICC), onde KA (Key Agreement) é o algoritmo de estabelecimento de chaves;

b) as chaves de sessão K_MAC = KDF_MAC(K) e K_ENC = KDF_ENC(K), onde KDF_MAC e KDF_ENC são funções derivadoras de chaves (Key Derivation Functions);

c) a chave pública do terminal comprimida Comp(PKe_IFD), que será utilizada no protocolo TA para a autenticação do terminal.

A Figura 3.11 ilustra o protocolo CAv1.

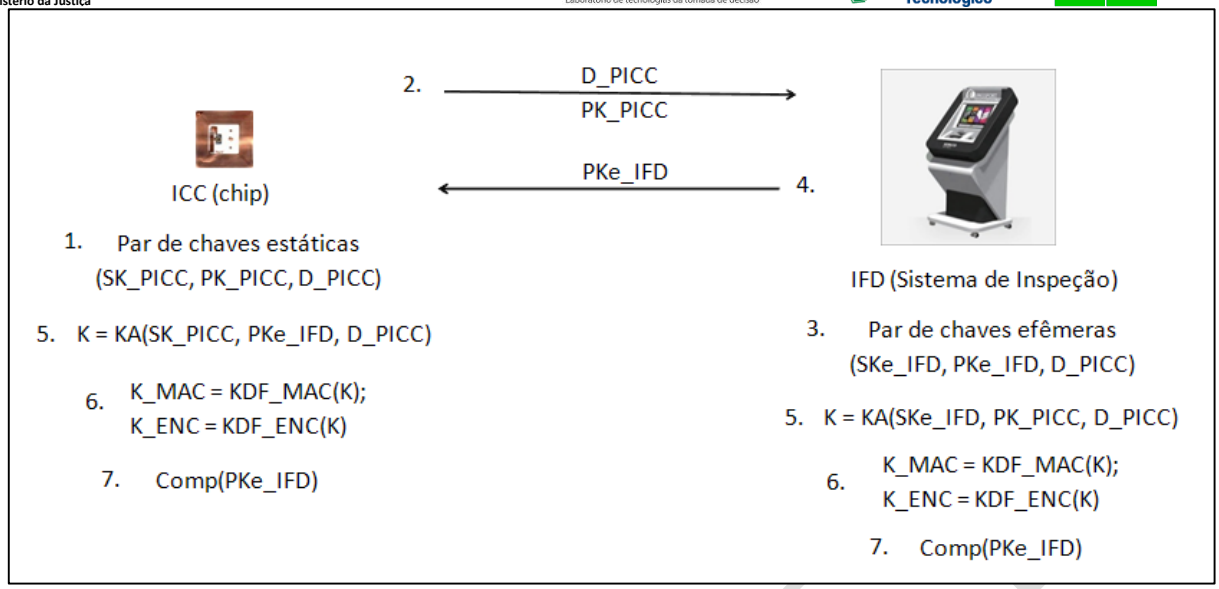

Figura 3.11: Protocolo CAv1.

Para verificar a autenticidade de PK_PICC, o terminal deve rodar o protocolo de Autenticação Passiva, PA, depois do protocolo CAv1. A chave pública do chip PK_PICC encontra-se no grupo de dados DG14 da LDS (ver Figura 3.3). O terminal pode acessar essa chave porque o protocolo de Controle de Acesso Básico BAC, ou o PACE (ver Seção 3.5), é rodado antes do EAC.

O protocolo **CAv1** provê uma **autenticação implícita do chip** e dos dados armazenados através do uso de canal seguro com as chaves de sessões obtidas. Portanto, esse protocolo é uma alternativa ao protocolo AA (padronizado pela ICAO), e conta com uma funcionalidade adicional, que é a geração de chaves de sessões robustas.

3.4.2. Autenticação do Terminal Versão 1 (TAv1)

O protocolo TAv1 funciona da seguinte forma.

1. O terminal envia uma cadeia de certificados para o chip, contendo o certificado do terminal, C_T, e o certificado de um Verificador de Documento, C_DV, pertencente a PKI da CVCA.

2. O chip, que contém o C_CVCA de seu país, primeiro usa a chave pública do CVCA (que se encontra no C_CVCA) para verificar a autenticidade de C_DV. Depois ele

usa a chave pública do DV (que se encontra no C_DV) para verificar C_T. E Finalmente, dado que C_T é autêntico, o chip extrai a chave pública do terminal PK_IFD.

3. O chip escolhe aleatoriamente um desafio r_PICC e envia para o terminal.
4. O terminal se autentica através da assinatura s_IFD = S[SK_IFD](ID_PICC || r_PICC || Comp(PKe_IFD)).
5. O chip então verifica se a assinatura do terminal é válida através do cálculo V[PK_IFD](s_IFD, ID_PICC || r_PICC || Comp(PKe_IFD)).

Figura 3.12 ilustra o protocolo TAv1.

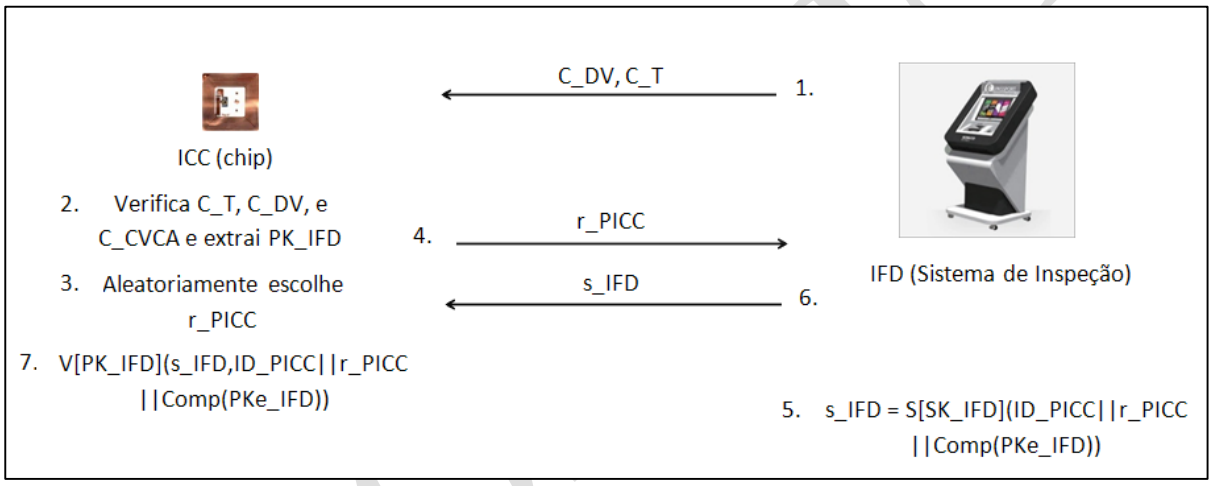

Figura 3.12: Protocolo TAv1.

A identificação do chip, ID_PICC, depende do mecanismo de controle de acesso usado. Se for o BAC, o ID_PICC será o número do documento incluindo o dígito verificador; se for o PACE, o ID_PICC será Comp(PKe_PICC), onde a chave pública efêmera do chip PKe_PICC é gerada durante o protocolo PACE.

O protocolo TAv1 também autentica a chave pública efêmera PKe_IFD escolhida pelo terminal durante o protocolo CAv1. O chip deve amarrar as permissões do terminal (no protocolo Secure Messaging executado depois do TAv1) com a chave PKe_IFD. É importante ressaltar que durante o protocolo TAv1, todas as mensagens devem ser transmitidas usando o protocolo Secure Messaging (SM) com as chaves de sessão obtidas do BAC ou PACE. Isso garante integridade e confidencialidade das mensagens. Uma

possível consequência do não uso do protocolo SM seria ataques de Denial of Service (Negação de Serviço), também chamado de DoS; um adversário poderia por exemplo enviar falsas assinaturas s' para o chip, e o chip iria interpretar que o terminal não tem a devida permissão de acesso aos dados sensíveis do chip.

3.4.3. Autenticação do Terminal Versão 2 (TAv2)

No mecanismo de Controle de Acesso EACv2, a autenticação do terminal (TAv2) é feita antes da autenticação do chip (CAv2). O protocolo TAv2 funciona da seguinte forma.

1. O terminal envia uma cadeia de certificados para o chip assim como no protocolo TAv1.
2. O chip usa o C_CVCA contido na sua memória para verificar a cadeia de certificados e então extrai a chave pública do terminal, PK_IFD.
3. O terminal gera um par de chaves efêmeras do tipo Diffie-Hellman PKe_IFD , SKe_IFD, e envia Comp(PKe_IFD) para o chip, junto com possíveis informações adicionais A_IFD.
4. O chip escolhe aleatoriamente um desafio r_PICC e o envia para o terminal.
5. O terminal se autentica através da assinatura s_IFD = S[SK_IFD](ID_PICC || r_PICC || Comp(PKe_IFD || A_IFD)).
6. O chip usa a chave pública do terminal, PK_IFD, para verificar se a assinatura s_IFD é autentica.

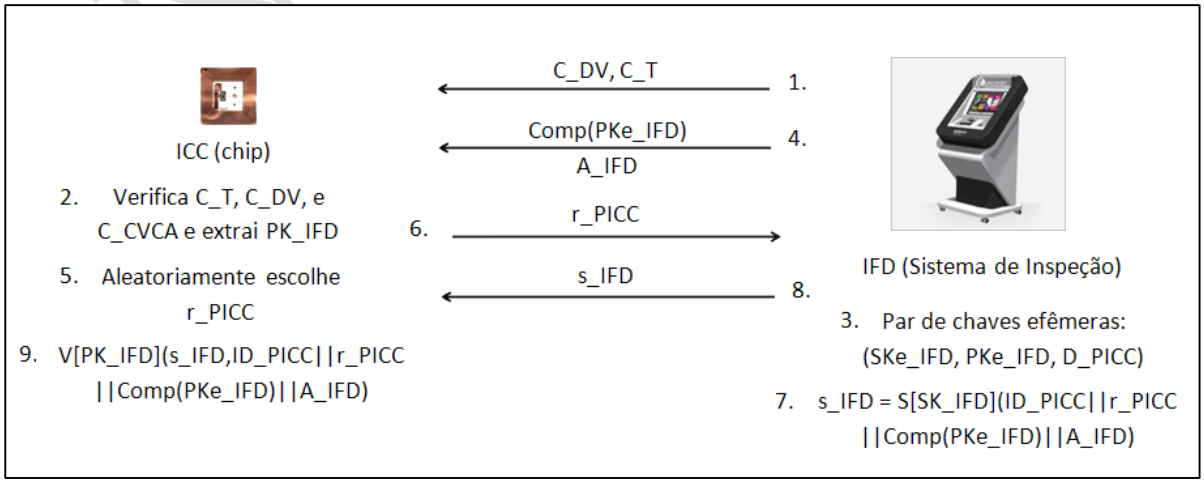

Figura 3.13: Protocolo TAv2.

O conjunto de parâmetros para estabelecimento de chaves Diffie-Hellman, D_PICC, usado pelo terminal para o cálculo do par de chaves efêmeras, é obtido durante o protocolo PACE (ver Seção 3.5), que é rodado antes do EAC.

O protocolo TAv2 também autentica uma chave pública efêmera, PKe_IFD, que será usada para configurar o canal seguro (usando o Secure Messaging) durante o protocolo de autenticação do chip, CAv2. Toda a comunicação entre o chip e o terminal depois do EACv2 deve ser protegida usando o protocolo SM, já que o terminal terá acesso a dados sensíveis do chip. Nesse protocolo, a identificação do chip, ID_PICC, assim como no protocolo TAv1, depende do tipo de mecanismo de controle de acesso usado, BAC ou PACE.

3.4.4. Autenticação do Chip Versão 2 (CAv2)

O protocolo CAv2 funciona da seguinte forma.

1. O chip envia para o terminal a sua chave pública (estática) PK_PICC e uma conjunto de parâmetros para estabelecimento de chaves Diffie-Hellman, D_PICC.
2. O terminal envia para o chip a sua chave pública efêmera, PKe_IFD, calculada durante a execução do protocolo TAv2.
3. O chip calcula uma versão comprimida de PKe_IFD, Comp(PKe_IFD) e compara com o valor que foi obtido anteriormente durante o protocolo TAv2.
4. Ambos, o terminal e o chip, calculam a chave simétrica secreta K = KA(SK_PICC, PKe_IFD, D_PICC) = KA(SKe_IFD, PK_PICC, D_PICC).
5. O chip escolhe aleatoriamente um nonce r_PICC, obtém as chaves de sessão K_MAC = KDF_MAC(K, r_PICC) e K_ENC = KDF_ENC(K, r_PICC) para uso no protocolo Secure Messaging, calcula um token de autenticação T_PICC = MAC(K_MAC, PKe_IFD), e manda r_PICC e T_PICC para o terminal.

6. Tendo obtido r_PICC, o terminal calcula K_MAC e K_ENC e usa K_MAC para validar o token T_PICC.

Figura 3.14 ilustra o protocolo CAv2.

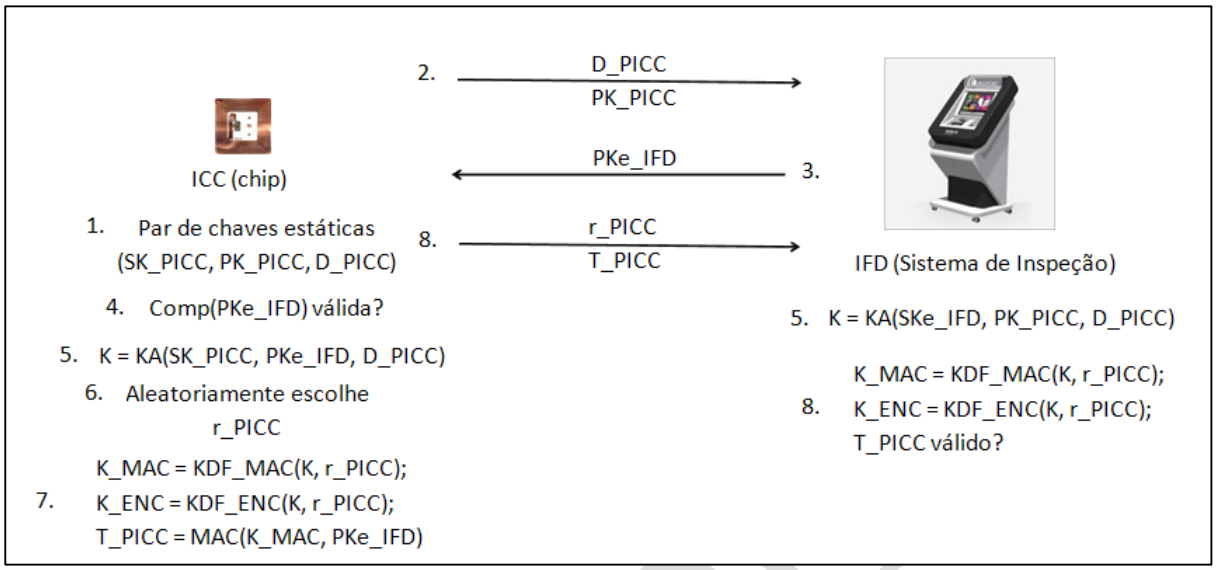

Figura 3.14: Protocolo CAv2.

A

Para verificar a autenticidade de PK_PICC, o terminal deve rodar o protocolo de Autenticação Passiva, PA, antes do protocolo CAv2.

Ao contrário do protocolo **CAv1**, o protocolo CAv2 provê uma **autenticação explícita do chip**, já que o terminal explicitamente verifica o token da autenticação enviado pelo chip. Assim como o CAv1, o CAv2 é uma alternativa ao protocolo AA e também conta com a funcionalidade adicional de geração de chaves de sessões robustas.

3.5. Controle de Acesso Suplementar (SAC)

Como visto na Seção 3.3, o nível de segurança do mecanismo de Controle de Acesso Básico, BAC, é limitado pelo uso exclusivo de criptografia simétrica. Isso significa que a entropia das chaves geradas no BAC, para cifrar e autenticar a comunicação sem fio entre o chip e o terminal, é retida pela baixa entropia do MRZ. Para superar essa fraqueza do BAC, em 2010 a ICAO especificou um Controle de Acesso Suplementar (SAC – Supplemental Access Control). O SAC se baseia num protocolo publicado pelo BSI em 2007, o PACE (Password Authenticated Connection Establishment) (ver [4,5]), que conta

com o uso de criptografia assimétrica para a geração de chaves de sessões simétricas. A criptografia assimétrica possibilita que a entropia das chaves geradas no PACE seja independente da entropia da chave de entrada (ou senha), isso sendo a principal vantagem do PACE em relação ao BAC.

No relatório técnico TR-SAC [3], a ICAO define várias opções de implementação para o PACE e requer que ele seja implementado além do BAC para que se tenha interoperabilidade entre países. A ICAO requer que caso o chip do passaporte eletrônico suporte o PACE, os Sistemas de Inspeção sempre escolham executar o PACE. No entanto, o IS nunca deve executar ambos o BAC e o PACE numa mesma sessão. De acordo com uma decisão da Comissão Europeia, a implementação do PACE nos passaportes de países europeus deve ser feita até o final de 2014 e o BAC coexistirá com o PACE por um tempo ainda indeterminado por questões de compatibilidade e interoperabilidade.

3.5.1. PACE

O PACE é um protocolo de estabelecimento de chaves do tipo Diffie-Hellman autenticado por uma senha, π, compartilhada entre o chip e o terminal. Além de o PACE ter um nível de segurança bem maior que a do BAC, ele adiciona flexibilidade, consentindo diferentes tipos de senhas: MRZinfo, CAN e PIN.

MRZinfo: essa senha é a 'Informação do MRZ', obtida da mesma forma que para o BAC. Ela deve ser obrigatoriamente suportada pelo PACE.

CAN: o CAN (Card Access Number), ou Número de Acesso do Cartão, é uma senha curta que pode ser impressa ou exibida no MRTD e que deve ser escolhida aleatoriamente ou pseudo-aleatoriamente. Uma das vantagens do CAN é que ele pode ser facilmente digitado manualmente, eliminando custos com leitoras ópticas. O CAN é opcional para o PACE.

PIN: o Número de Identificação Pessoal (PIN – Personal Identification Number) é uma senha curta que só deve ser conhecida pelo legítimo titular do documento. O PIN pode ser usado, por exemplo, para acessar a aplicação de eID no caso de cartões eletrônicos -- ele não é usado para acessar aplicações de controle de fronteira. O PIN é de uso obrigatório por todos os terminais de autenticação e é o único tipo de senha usado no PACE que pode ser bloqueado depois de algumas tentativas de autenticação sem sucesso. Caso o PIN seja

bloqueado, usa-se a senha PUK (PIN Unblock Key) para acessar um mecanismo de desbloqueio do PIN.

Note que enquanto o MRZinfo é uma senha estática, o CAN e o PIN são senhas que podem ser alteradas.

De uma maneira resumida, o PACE funciona da seguinte forma.

Configuração do protocolo: primeiramente, o Sistema de Inspeção deve ler um arquivo contido no chip do MRTD, chamado CardAccess -- o acesso de leitura a esse arquivo não é restrito. O CardAccess contém informações se o chip suporta o PACE e caso positivo ele provê parâmetros a serem usados na execução do protocolo. Esses parâmetros incluem a cifra simétrica, os algoritmos de estabelecimento de chaves, os parâmetros do domínio, e funções de mapeamento que veremos a seguir. (Se o arquivo CardAccess não estiver disponível no chip, o IS deve ler o MRTD com o BAC.) Em se tratando de passaportes eletrônicos, o IS deve obter uma chave K_π a partir do MRZinfo ou do CAN: K_π = KDF_π(MRZinfo) ou K_π = KDF_π(CAN), onde KDF é uma função derivadora de chaves.

Execução do protocolo: a execução do protocolo entre o chip e o terminal é composta de quatro etapas, a saber.

1. O chip aleatoriamente escolhe um nonce s, cifra s com K_π e envia o texto-cifrado para o terminal, onde é decifrado com a chave K_π. A cifragem deve ser feita no modo CBC, de acordo com ISO/IEC 10116, e com IV=0.
2. Uma função de mapeamento, Map, é usada por ambos o chip e o terminal para mapear em parâmetros efêmeros de criptografia assimétrica, tipicamente um gerador secreto de grupo.
3. O chip e o terminal executam um protocolo de estabelecimento de chaves do tipo Diffie-Hellman baseado nos parâmetros obtidos no item 2.
4. Chaves de sessão são obtidas e confirmadas através de tokens de verificação.

Troca Segura de Mensagens (Secure Messaging): após a execução do PACE, o

IS pode então acessar dados menos sensíveis do chip, por exemplo, informações contidas no MRZ, foto do rosto do portador do MRTD, informações sobre a chave pública de autenticação ativa, etc., e pode também acessar o Objeto de Segurança do chip, SO_d.

A Figura 3.15 ilustra o protocolo PACE de forma mais detalhada.

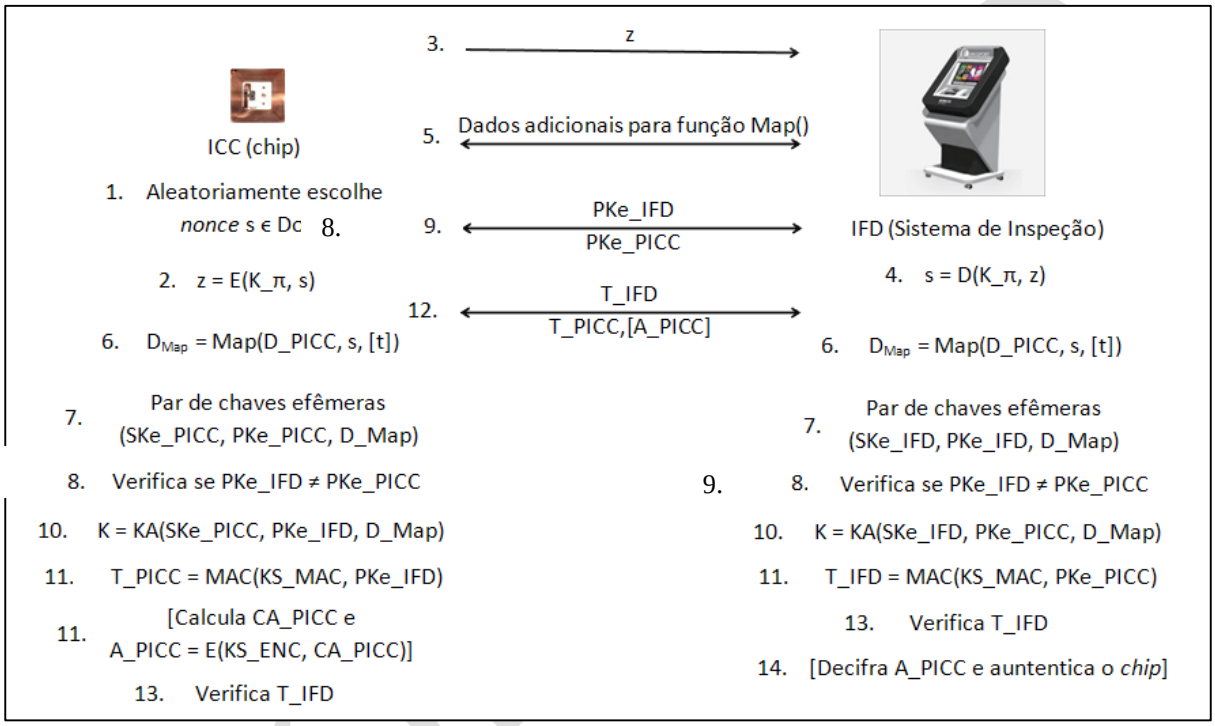

Figura 3.15: Protocolo PACE.

Como parte do protocolo de execução, o PACE oferece várias funções de mapeamento, Map. Cada uma delas pode ser implementada tanto com curvas elípticas como com criptografia padrão (método de estabelecimento de chaves assimétrico). As funções Map podem ser de três tipos, a saber.

Mapeamento Genérico: baseado num sistema de estabelecimento de chaves do tipo Diffie-Hellman, é baseado em operações genéricas de grupos e pode ser facilmente implementado.

Mapeamento Integrado: baseado num mapeamento direto de um elemento de campo para o grupo criptográfico, é ligeiramente mais rápido que o Mapeamento Genérico. Na Figura 3.15, os colchetes, [ ], representam parâmetros ou passos adicionais a serem

executados no PACE caso o Mapemeamento Integrado seja usado.

Mapeamento com Autenticação do Chip. estende o Mapeamento Genérico e integra autenticação do chip no protocolo PACE.

A primeira versão do PACE e do PACEv1 define somente o mapeamento genérico e está obsoleta. Ela evoluiu para o PACEv2, que define uma extensão do Mapeamento Genérico, o Mapeamento com Autenticação do Chip, e o Mapeamento Integrado.

O funcionamento da função Map é exemplificado aqui com o Mapeamento Genérico usando DH (sem o uso de curvas elípticas): a função Map: g→g_Map é definida como g_Map=g^s.h, onde o elemento h ϵ <g> deve ser calculado como h = KA(SKmap_PICC,PKmap_IFD,D_PICC)=KA(SKmap_IFD,PKmap_PICC,D_PICC). Para isso, o chip e o IFD (Sistema de Inspeção) usam os parâmetros de domínio estáticos do chip D_PICC para gerar pares de chaves (SKmap_PICC,PKmap_PICC) e (SKmap_IFD,PKmap_IFD), respectivamente. Note que o IFD obteve D_PICC através da leitura do arquivo CardAccess. Na troca de dados adicionais (item 5 na Figura 3.15), o chip e o IFD trocam PKmap_PICC e PKmap_IFD (chaves públicas efêmeras), e podem então calcular h. Vale salientar que o ataque de Man in the Middle (MitM) não é um problema aqui. Se um adversário executar o ataque MitM, as chaves h obtidas pelo chip e pelo IFD serão diferentes. Isso significa dizer que os parâmetros efêmeros de domínio D_Map=Map(D_PICC,s) serão diferentes para o chip e o IFD, o que vai resultar numa falha do protocolo PACE.

Para o estabelecimento de chaves do tipo Diffie-Hellman, tanto o chip como o IFD geram pares de chaves efêmeras baseadas nos parâmetros efêmeros do domínio, D_map, obtidos através da função Map, e fazem o intercâmbio de suas chaves públicas efêmeras PKe_IFD, PKe_ICC. Eles devem verificar se as duas chaves são diferentes, e, caso positivo, podem gerar a chave simétrica K = KA(SKe_PICC,PKe_IFD,D_Map) = KA(SKe_IFD,PKe_PICC,D_Map). Novamente, um adversário que tente realizar um ataque MitM na troca de chaves públicas efêmeras não terá sucesso. Isso porque o adversário não conhece D_Map e portanto não conseguirá obter uma chave simétrica com o chip e outra com o IFD. Note que para obter D_Map, o adversário teria que conhecer o nonce s, que por

conseguinte só pode ser com o conhecimento de K_π.

As chaves de sessão KS_MAC = KDF_MAC(K) e KS_ENC(K) = KDF_ENC(K) são obtidas da mesma forma que para o BAC. O chip e o IFD trocam tokens de autenticação T_IFD = MAC(KS_MAC,PKe_PICC) e T_PICC = MAC(KS_MAC,PKe_IFD) provando que têm a chave KS_MAC.

Na Figura 3.15, o item 11 em colchetes, e o item 14 só são executados se o Mapeamento Integrado for usado. Esses passos servem para que o terminal verifique a autenticidade do chip.

É importante ressaltar que a segurança do PACE depende fortemente do sigilo da senha π e diferentemente do BAC, o PACE não está sujeito a ataques de força-bruta. Note que no PACE um adversário pode interceptar o canal de comunicação do e-passport para a leitora e obter z. No entanto, como o adversário não tem o nonce s, ele não tem um par de texto claro e texto cifrado para fazer criptanálise da chave K_π usada para a cifragem de s. Além disso, a entropia da chave simétrica K gerada no estabelecimento de chaves Diffie-Hellman não depende da entropia da senha. Por outro lado, se um adversário tomar conhecimento da chave K_π, ele poderá realizar o MitM na troca de chaves públicas efêmeras. Para isso ele lê D_PICC, decifra z usando K_π, calcula D_Map e depois calcula uma chave K com o chip e uma com o Sistema de Inspeção. Por razões técnicas, somente senhas estáticas são usadas no protocolo PACE. Tipicamente essa senha é impressa no documento e um adversário uma vez tendo visto o MRTD, sabe a senha estática. Em [10], Ullmann e Vögeler discutem sobre o uso de displays flexíveis que permitem aplicar senhas dinâmicas em protocolos baseados em senhas, como o PACE. Se senhas dinâmicas forem aplicadas a cada vez que o PACE for rodado, um adversário não terá nenhuma vantagem em saber uma senha passada. Uma das grandes vantagens do protocolo PACE é que ele oferece forte forward secrecy. Isso quer dizer que se uma senha é comprometida, chaves de sessão K previamente estabelecidas e apagadas após a execução do protocolo permanecem seguras. Ou seja, as comunicações anteriores entre o chip e o IFD não são comprometidas.

3.6. Fluxo de Execução dos Protocolos

De acordo com a ICAO, o Sistema de Inspeção deve seguir os seguintes passos para a execução dos protocolos de autenticação.

1. Ler o arquivo CardAccess (RECOMENDADO). Ao fazer isso, o Sistema de Inspeção saberá quais mecanismos de controle de acesso e autenticação são suportados pelo chip do e-passport e quais parâmetros devem ser utilizados. Esse arquivo pode não estar disponível se o PACE não for suportado.

2. Executar o PACE (RECOMENDADO se o PACE for suportado pelo chip). O chip aceita como senhas o MRZ (OBRIGATÓRIO) e o CAN (OPCIONAL). Se o protocolo for executado com sucesso, o protocolo de troca segura de mensagens, Secure Messaging, deve ser iniciado e então o chip permite que o Sistema de Inspeção tenha acesso a dados menos sensíveis do chip. Esses dados incluem dados dos grupos de dados DG1, DG2, DG14 (contendo a chave pública para execução do protocolo de Autenticação do Chip, CA), DG15, etc., e também o objeto de segurança do documento, SO_d.

3. Executar o BAC (CONDICIONAL). É recomendado que o BAC seja executado caso o PACE não tenha sido executado, e que o protocolo Secure Messaging seja iniciado caso o BAC tinha sido rodado com sucesso. O chip concede acesso aos mesmos dados concedidos no PACE.

4. Executar o protocolo de Autenticação Passiva, ou PA (OBRIGATÓRIA) para verificar a autenticidade dos dados contidos no chip. A assinatura de SO_d é verificada. Nesse passo, o Sistema de Inspeção também deve comparar as informações de segurança (SecurityInfos) contidas no arquivo CardAccess com informações contidas no grupo de dados DG14.

5. Executar o protocolo de Autenticação Ativa, ou AA (OPCIONAL) para verificar que os dados foram lidos de um chip autêntico e que o chip não foi substituído. O chip deve conter o grupo de dados DG15 com informações da chave pública de autenticação ativa. O IS deve ser capaz de ler o MRZ.

6. Executar o EAC (OPCIONAL). Quando executado com sucesso, o chip concede acesso a dados mais sensíveis, tais como imagens de impressões digitais ou íris, a um Sistema de Inspeção autorizado a ler tais dados.

O BSI descreve dois procedimentos de inspeção ligeiramente diferentes. O procedimento de inspeção padrão é para documentos compatíveis com a ICAO que não suportam o EAC, e/ou para terminais que não suportam o EAC. O procedimento de inspeção avançado pode ser usado por terminais que suportam o EAC em documentos que também suportam o EAC.

O procedimento de inspeção padrão consiste dos seguintes passos.

1. Leitura do arquivo CardAccess.
2. Execução do PACE/BAC; Permissão de acesso a dados menos sensíveis usando o Secure Messaging.
3. Início do protocolo PA. A assinatura do SO_d é verificada, incluindo a validação do certificado.
4. Execução do protocolo AA.
5. Leitura e autenticação dos dados; Fim do protocolo PA. Os hashes dos grupos de dados são comparados com os do SO_d.

O procedimento de inspeção avançado consiste dos seguintes passos.

1. Leitura do arquivo CardAccess.
2. Execução do PACE/BAC; Permissão de acesso a dados menos sensíveis usando o Secure Messaging.
3. Execução do protocolo CA (parte do EACv1); Reinicialização do protocolo Secure Messaging.

4. Início do protocolo PA. A assinatura do SO_d é verificada, incluindo a validação do certificado.

5. Execução do protocolo AA (opcional).

6. Execução do protocolo TA (parte do EACv1); acesso a dados mais sensíveis, como os dos grupos de dados DG3 (imagem das impressões digitais) e DG4 (imagens das íris) é concedido.

7. Leitura e autenticação dos dados; Fim do protocolo PA. Os hashes dos grupos de dados são comparados com os do SO_d.

No procedimento de inspeção avançado, a Autenticação Ativa é opcional já que o protocolo de Autenticação do Chip (CA) é executado, garantindo a autenticidade do chip. Os procedimentos de inspeção definidos pelo ICAO e pelo BSI são detalhados em [14].

4. ANÁLISE DOS MECANISMOS DE AUTENTICAÇÃO USADOS EM E-PASSPORTS

Esse capítulo se concentra na análise dos mecanismos de autenticação usados em passaportes eletrônicos. Ele visa responder as seguintes perguntas.

1) Quais as funções desses mecanismos/protocolos?

2) Que tipos de dados precisam ser armazenados no chip ou no Sistema de Inspeção?

3) O chip necessita de capacidade de processamento?

4) Que vulnerabilidades podem ser encontradas nos mecanismos de autenticação? Elas podem ser evitadas?

A Tabela 4.1 ilustra os tipos de mecanismos de autenticação descritos na Seção 3, mostrando de forma resumida suas funcionalidades e o tipo de método utilizado.

Abreviação	Nome do protocolo	Função	Método
BAC	Basic Access Control.	Proteger o chip contra-ataques de skimming e proteger a comunicação entre o chip e o terminal contra-ataques de eavesdropping.	Desafio-resposta e leitura ótica (criptografia simétrica).
PACE	Password Authenticated Connection Establishment.	Proteger o chip contra-ataques de skimming e eavesdropping.	Estabelecimento de chaves Diffie-Hellman autenticado por senha compartilhada.
EAC	Extended Access Control.	Consiste dos protocolos CA e TA. Previne acesso não autorizado a dados biométricos adicionais (dados sensíveis como impressão digital e íris).	Criptografia assimétrica.
	CA: Chip Authentication.	Estabelecimento de um canal de comunicação seguro e deteção de chips clonados.	Estabelecimento de chaves Diffie-Hellman.
	TA: Terminal Authentication.	Autenticação de terminal para acesso a dados sensíveis do chip.	Desafio-resposta.
PA	Passive Authentication.	Verificação da autenticidade e integridade dos dados contidos no chip.	Assinatura digital off-line.
AA	Active	Deteção de clonagem e	Desafio-resposta

	Authentication.	substituição de chips.	e leitura ótica (criptografia assimétrica).

Tabela 4.1: Funções dos mecanismos de autenticação

Na Tabela 4.2 analisamos que tipos de dados extras precisam ser armazenados no chip de um passaporte eletrônico para que um determinado mecanismo de autenticação possa ser executado. Essa tabela também identifica se o chip requer capacidade de processamento ou não.

Protocolo		Dados armazenados no *chip*	Capacidade de processamento do *chip*?
BAC		K_ENC, K_MAC	Sim
PACE		Parâmetros inclusos no *CardAccess*	Sim
EAC	**CA**	SK_PICC, PK_PICC	Sim
	TA	C_CVCA	Sim
PA		SO_d (com C_DS opcional)	Não
AA		KPr_AA, KPu_AA (armazenados no SO_d)	Sim

Tabela 4.2: Dados extras armazenados no chip do MRTD e necessidade de capacidade de processamento do chip.

No caso da Autenticação Passiva (PA), o chip não requer capacidade de processamento. No entanto, o Sistema de Inspeção precisa ter acesso a uma cadeia de certificados da PKI da CSCA. Certificados raízes C_CSCAs de diversos países são armazenados no sistemas de inspeção de cada país participante da PKI, enquanto que certificados de Assinantes de Documentos, os C_DS, contendo a chave pública dos DS´s podem ser obtidos através da ICAO-PKD.

O protocolo AA, por sua vez, necessita que o chip tenha capacidade de processamento, já que o chip precisa assinar um desafio enviado pelo Sistema de Inspeção. Todavia, o IS não precisa de nenhuma informação adicional além daquelas que já são acessíveis no chip.

Em se tratando do BAC, o chip também requer capacidade de processamento; durante o protocolo desafio-resposta o chip precisa, por exemplo, usar algoritmos criptográficos para autenticar o Sistema de Inspeção. O chip armazena as chaves de

Acesso Básico K_ENC e K_MAC de forma segura, para que não precise calculá-las durante o protocolo. Aqui o Sistema de Inspeção também não precisa de informações adicionais para a execução do protocolo. O BAC não requer uma PKI.

Para que possa executar o protocolo PACE, o chip precisa armazenar o arquivo CardAccess contendo um conjunto de parâmetros, tais como: a cifra simétrica, os algoritmos de estabelecimento de chaves, os parâmetros do domínio, e funções de mapeamento. Durante o protocolo, o chip precisa processar várias informações, como por exemplo a cifragem de um nonce e os cálculos para o estabelecimento de chaves Diffie-Hellman.

Por ter que, respectivamente, calcular uma chave simétrica e verificar assinaturas digitais, o chip usado no protocolos CA e TA necessitam de capacidade de processamento. No caso do CA, o chip deve ter armazenado em sua memória um par de chaves do tipo Diffie-Hellman para a autenticação do chip, enquanto que no caso do TA o chip precisa ter gravado em sua memória o certificado raiz da CVCA para verificar a chave pública do terminal e autorizar ou não o acesso a dados sensíveis do chip.

A Tabela 4.3 mostra algumas das fraquezas/vulnerabilidades dos mecanismos de autenticação.

Protocolo		Fraquezas
BAC		Baixa entropia do MRZ.
PACE		-
EAC	CA	-
	TA	Ausência de relógio interno no chip; requer gerenciamento adicional de chaves.
PA		Não deteta clonagem/substituição de chips.
AA		Possibilidade de rastreamento.

Tabela 4.3: Fraquezas dos mecanismos de autenticação

Como visto anteriormente, a Autenticação Passiva (PA) embora seja muito eficaz

para detectar se os dados contidos no chip do passaporte foram manipulados, ela não detecta se o chip foi clonado ou substituído. Essa detecção pode ser feita, por exemplo, pela Autenticação Ativa que verifica se o chip e a página de dados do passaporte pertencem um ao outro.

Já a Autenticação Ativa (AA), apesar de ser um protocolo simples e eficiente na detecção de chips clonados, introduz uma ameaça à privacidade: o rastreamento de chips. Isso se dá devido a semântica do desafio enviado pelo terminal [4 – Apêndice B (Challenge Semantics)]. Na seção 3.2 (Autenticação Ativa) consideramos que o IFD envia um nonce, o RND.IFD, aleatório para o chip. Nesse caso, o chip assina uma mensagem sem saber a semântica dessa mensagem; a Autenticação Ativa pode ser usada para que o chip assine qualquer mensagem que o terminal queira. O terminal pode, portanto, ao invés de gerar um nonce aleatório, gerar uma sequência de bits imprevisível, que possa ser publicamente verificada; o IFD possui um par de chaves pública e privada, KPu_IFD e KPr_IFD, e usa sua chave privada KPr_IFD para assinar um conjunto de dados contendo a identidade do chip, a data, a hora e o local em que o chip se encontrava no momento de execução do protocolo desafio-resposta. O novo desafio se torna:

$$c = S[KPr\_IFD](ID\_PICC \,||\, Data \,||\, Hora \,||\, Local).$$

Com isso, o Sistema de Inspeção pode usar a assinatura do chip para provar que o chip esteve de fato num certo dia, horário e local, ou seja, o IFD pode usar a assinatura do chip para provar que uma pessoa realmente imigrou. A prova é válida porque terceiros, que confiam no IFD e que conhecem sua chave pública, podem verificar a assinatura ‘c’. O problema é que, como a Autenticação Ativa não é restrita somente a terminais autorizados, essa prova pode ser usada de má forma para rastrear pessoas. O pior cenário é quando a AA é usada sem o BAC. Nesse caso, o chip pode ser acessado a distância por terminais não autorizados pelo dono do MRTD, e um poderoso sistema de rastreamento pode ser montado. O BAC diminui o problema já que força uma interação visual entre o Sistema de Inspeção e o documento.

O principal defeito do mecanismo BAC é que as chaves de Acesso Básico são obtidas a partir da ‘informação do MRZ’, e essa informação não pode ser trocada. Embora

as chaves K_ENC e K_MAC tenham 16 bytes cada, as entropias dessas chaves são bem menores. A entropia de uma chave BAC é idêntica a entropia da 'informação do MRZ' já que essa informação é a única fonte de entropia no cálculo da chave. Essa entropia pode chegar a um valor tão baixo quanto 35 bits se um atacante tiver uma ideia da data de nascimento do portador do MRTD, e se for feita uma correlação entre a data de expiração do documento e o número do documento. Portanto, um ataque de força-bruta pode ser facilmente realizado para obter as chaves BAC. Para isso basta que um adversário obtenha pares de texto claro e texto cifrado através da interceptação dos canais de comunicação entre o chip e o IFD. Embora a interceptação da comunicação nos controles de fronteira não seja tão fácil devido a necessidade de um interceptador estar relativamente próximo (alguns metros) da leitora e do e-passport, isso não é uma limitação caso os e-passports sejam lidos em bancos, hotéis ou companhias aéreas, por exemplo. Em [9,11] os autores discutem contramedidas paliativas para o BAC, tais como: a) O aumento progressivo e limitado no tempo de resposta do chip, prolongando o tempo que um adversário precisa interceptar a comunicação; b) A introdução de números alfanuméricos aleatórios juntamente com um dígito verificador na "Informação do MRZ", para aumentar a entropia da chave K_π (O dígito verificador é necessário porque os campos do MRZ deve ser lidos corretamente pela leitora óptica; qualquer erro na leitura causará erro no sistema.). c) O uso de números aleatórios de passaportes, ao invés do uso de um sistema determinístico na geração dos números; d) O uso da uma capa bloqueadora de rádio frequência, ou gaiola de Faraday, como é feito nos Estados Unidos. Notoriamente, nenhuma dessas contramedidas resolve a deficiência do BAC, que é a sua dependência na entropia do MRZ.

Como para a aplicação offline não é possível a consulta pelo chip a LCR dos terminais, os certificados dos terminais são emitidos com uma curta validade. Uma vulnerabilidade no protocolo TA se dá pelo fato de chips de MRTDs usados atualmente não possuírem relógio interno Isso impossibilita que o chip verifique se o certificado do terminal, C_T, expirou/foi revogado ou não. Inicialmente a data armazenada no chip é a data da fase de pré-personalização do chip e é então atualizada, por exemplo, com a data do certificado mais recente de um C_CV válido.

Já o CA, apesar de ser um protocolo para a autenticação do chip, assim como o AA, ele não é susceptível a mesma fraqueza que o AA, ou seja, a ataques de semântica do

desafio para rastreamento do chip; uma vez que a prova de autenticação do chip é baseada num protocolo de estabelecimento de chaves Diffie-Hellman, ela não é transferível para terceiros.

O PACE foi introduzido para substituir o BAC. Ao contrário deste, que usa criptografia simétrica para facilitar implementação, o protocolo PACE é baseado em criptografia assimétrica e a robustez das chaves de sessões produzidas é independente da entropia da senha usada. O tamanho das senhas pode ser tão curto como 6 dígitos, como é o caso do CAN. Recentemente, Deufel et al. [12] propuseram um protocolo chamado de BioPACE, como sendo uma etapa de pré-processamento para o protocolo PACE. A ideia do BioPACE é a de substituir a chave baseada numa senha compartilhada por uma chave baseada na biometria do titular do MRTD, e tem como proposito substituir o EAC e a PKI relacionada, ou seja, a PKI do CVCA. Em [13] os autores fornecem uma ampla avaliação do BioPACE no que diz respeito a recursos de segurança. Eles documentam algumas fraquezas do BioPACE se comparado ao PACE, em especial que o protocolo BioPACE permite rastreamento do chip e limita a flexibilidade do controle de acesso; o BioPACE só oferece dois níveis de autorização: autorização para ler todos os grupos de dados, ou não ler nenhum grupo de dado do chip. Por outro lado, o EAC oferece um controle de acesso mais refinado. Portanto, o PACE em conjunto com o EAC são os protocolos recomendados para um bom controle de acesso.

5. A IDENTIDADE ELETRÔNICA ALEMÃ (nPA)

A Alemanha lançou seu novo cartão de identificação pessoal (neuer Personalausweis - nPA) em novembro de 2010. O nPA, ilustrado na Figura 5.1, é um cartão de policarbonato que tem o formato ID-1, do tamanho de um cartão de crédito, e contém um chip sem contato. O chip se comunica com um terminal que atua como um dispositivo de leitura ou escrita. Uma das vantagens de se ter um chip integrado no documento é que ele melhora a proteção contra falsificação e permite a inclusão de dados biométricos para fortalecer o elo entre o cartão e o seu titular. Além disso, o documento pode ser usado não só como documento de identificação visual, mas também para funções eletrônicas online ou offline.

Figura 5.1: Representação visual, frente e verso, do nPA.

O nPA pode ser usado como documento de viagem para uso governamental (função ePass), autenticação eletrônica mútua para aplicações como eGovernment ou eBusiness (função eID), e para a geração de assinaturas eletrônicas qualificadas (QES – Qualified Electronic Signature), que pela lei alemã, têm a mesma validade que assinaturas feitas à mão. Essas assinaturas podem ser usadas, por exemplo, para declaração de impostos.

Os principais objetivos de design do nPA foram os descritos abaixo.

- **Segurança de dados:** se baseia em algoritmos criptográficos e na resistência do chip contra adulterações.
- **Privacidade:** o nPA é a solução de eID que mais preserva a privacidade do cidadão. O nPA usa funções de minimização de dados, em que o chip só envia dados estritamente necessários para uma determinada transação online, e usuários não são vinculados a transações realizadas com provedores de serviços de domínios diferentes. Além disso, dados biométricos armazenados no chip (ex.: foto do rosto, impressões digitais) só podem ser acessados por entidades soberanas, tais como autoridades policiais, administração

aduaneira, unidades de investigação de impostos, e autoridades de registro. Esse acesso não pode ser feito na Internet; o cartão deve estar fisicamente presente. Mais ainda, a lei alemã não permite a criação de uma base central de dados com informações pessoais dos cidadãos, e os provedores de serviços não têm provas criptográficas que os dados acessados por ele pertencem a um determinado usuário.

- **Controle do usuário:** os dados transmitidos pelo chip para um provedor de serviços são controlados pelo usuário. Só são transmitidos para o provedor de serviços os dados autorizados pelo usuário, e somente se o provedor de serviços tiver permissão de acessar tais dados.

Para os indivíduos que não são cidadãos alemães, mas que tem permissão de residir no país, a Alemanha lançou em setembro de 2011 a permissão eletrônica de residência (elektronischer Aufenthalstiel - eAT), que assim como o nPA também possui funções eletrônicas.

Funções eletrônicas do nPA: o nPA combina o cartão de identidade convencional com três funções/aplicações eletrônicas:

função ePass: é uma função obrigatória do nPA, ou seja, o titular do cartão não pode escolher que essa função seja desativada. Ela é de uso exclusivo de autoridades soberanas (ex.: controle de fronteira), e permite que o nPA seja usado como um documento de viagens, sendo válido como um passaporte eletrônico substituto dentro da União Europeia. Uma diferença fundamental entre o nPA e o passaporte eletrônico alemão é que no caso do nPA, o armazenamento das impressões digitais dos dois dedos indicadores é voluntário;

função eID: esta função é opcional, mas não tem custos adicionais se o usuário escolher que essa função seja habilitada no momento do recebimento do cartão. Ela é usada para autenticação online ou offline, e pode ser usada, por exemplo, para aplicações de eBusiness, e eGovernment, ou em máquinas de vendas. A função eID oferece ao usuário do cartão a oportunidade de selecionar os dados que serão acessados por um provedor de serviços. Ela também permite o uso de pseudônimos para usar serviços de Internet, de forma a preservar a privacidade do usuário;

função eSign: essa função é opcional e só pode ser ativada pagando-se um valor adicional. A função eSign é usada para a geração de assinatura eletrônica qualificada, QES. Por lei, uma QES tem a mesma validade que uma assinatura feita à mão. Uma QES pode ser usada por exemplo, para a declaração de impostos de renda.

Os dados armazenados no chip são organizados de acordo com a função eletrônica,

ou seja, ePass, eID ou eSign, como será visto nas seções 5.2, 5.3, e 5.4. Acesso a qualquer dado só é possível depois de uma autenticação bem sucedida do terminal através do Procedimento de Autenticação Genérico, como será visto na Seção 5.1 (Acesso aos Dados do Cartão).

Mecanismos de autenticação usados pelo nPA: os seguintes protocolos criptográficos devem ser implementados pelo chip e terminal.

- BAC: é suportado no nPA por questões de compatibilidade, mas restrito a aplicações soberanas, ou seja, a função ePass.

- PACE: é implementado em todas as eIDs alemãs para proteger a comunicação sem fio entre o chip e a leitora de cartão local do usuário. Em se tratando de controle de fronteira, a leitora de cartão é o próprio terminal.

- EAC (TA e CA), versão 2: oferece um estabelecimento seguro de chaves e autenticação mútua entre o chip e o terminal. A leitora e o terminal devem estar conectados de maneira segura através do protocolo SSL/TLS antes de as chaves de sessão do protocolo EAC serem obtidas. A chave de sessão é usada no protocolo Secure Messaging. Por questões de privacidade, o par de chaves Diffie-Hellman de autenticação do chip, usado no protocolo CA, não é único. Um conjunto de cartões possuem o mesmo par de chaves, o que cria um conjunto de anonimato dentro daquele conjunto.

- PA: serve para provar a autenticidade dos dados armazenados no chip. Para isso, os dados armazenados na função ePass e a chave pública do chip são assinados usando a PKI da CSCA. Os grupos de dados não-oficiais da função eID não são assinados. Isso previne que um provedor de serviços passe dados para terceiras partes com prova criptográfica da autenticidade de que os dados foram lidos do chip.

- Um protocolo opcional chamado de Identificação Restrita (Restricted Identification - RI) oferece um método para usar pseudônimos de forma que a) eles possam ser vinculados por provedores de serviços individuais (domain-specific linkability), mas b) não entre diferentes provedores de serviços (cross-domain anonymity).

Senhas do nPA: são usadas para o protocolo PACE e podem ser diferentes, dependendo da aplicação (ver Figura 5.3):

- CAN: o Número de Acesso do Cartão (Card Access Number) é uma senha curta (6 dígitos) impressa ou exibida no cartão. Essa senha pode ser usada, por exemplo, para serviços de alteração ou em casos excepcionais como numa blitz policial.

- PIN: o Número de Identificação Pessoal (Personal Identification Number) é uma senha secreta curta (6 dígitos) que só deve ser conhecida pelo titular do cartão. Ou seja, essa senha é escolhida pelo próprio titular do cartão. Para isso, ele usa um outro PIN, o PIN de transporte, gerado durante o processo de fabricação do cartão e enviado para o titular do cartão através de uma carta. Essa senha é usada por exemplo para autenticação online.

- PUK: a chave de desbloqueio do PIN (PIN Unblock Key) é uma senha secreta longa (10 dígitos) que só deve ser conhecida pelo titular do cartão.

- Senha-MRZ: é uma chave secreta obtida da zona legível por máquina. Essa senha é usada, por exemplo, no caso de o cartão ser usado como documento de viagem no controle de fronteira.

Tipos de leitoras: o documento TR-03119 do BSI [25] descreve três categorias de leitoras de cartão que podem ser usadas com o nPA: Cat-B (Basic), Cat-S (Standard), e Cat-K (Komfort), ilustradas na Figura 5.2.

Figura 5.2 (tirada de [26]) – Leitoras *Basic*, *Standard*, *Komfort*.

- Leitora Basic (Cat-B): as leitoras dessa categoria não possuem PIN pad nem display. O PIN deve ser inserido através do teclado do computador. A leitora Basic pode ser usada para uso doméstico ou pode ser integrada com outros aparelhos tais como latptops. Ela pode ser usada para os serviços de eGovernment, verificação de idade, eTicketing, compras online, e outros. Apesar de esse tipo de leitora ter a vantagem de ser de baixo custo, ele tem a desvantagem de precisar que o usuário mantenha o computador protegido contra malwares, de forma que o PIN não seja obtido através de, por exemplo, keyloggers.

- Leitora Standard (Cat-S): possuem pelo menos o PIN pad. O PIN deve ser obrigatoriamente inserido através do PIN pad e nunca através do teclado do computador. Leitoras da categoria Cat-S devem ser usadas para aplicações com maiores requisitos de segurança.

- Leitora Komfort (Cat-K): possuem um PIN pad para permitir a inserção do PIN com segurança e um display de 2 x 16 caracteres alfa-numéricos. Leitoras da categoria Komfort suportam todas as funções do nPA, incluindo assinatura eletrônica qualificada. Para essas leitoras, a requisição da entrada do PIN, assim como os dados para o certificado de autorização do terminal devem aparecer no display. Além disso, o PIN deve ser obrigatoriamente inserido através do PIN pad.

É importante salientar que quando uma leitora com PIN pad é usada, o PACE é executado diretamente na leitora do cartão. Obviamente, esse tipo de leitora deve ser capaz de usar algoritmos de criptografia.

Tipos de terminais: o documento TR-03127 do BSI [24] descreve quatro tipos de terminais suportados pelo nPA: sistemas de inspeção; terminais de autenticação; terminais de assinatura confirmada; e terminais não autenticados, ilustrados na Figura 5.3.

- Sistemas de inspeção: podem ser do tipo nacional oficial ou do tipo estrangeiro oficial. Eles são terminais usados para verificação da aplicação da lei. Pode-se citar como exemplo os terminais de controle de fronteira, e autoridades investigadoras de impostos.

- Terminais de autenticação: podem ser do tipo doméstico oficial ou do tipo estrangeiro/não oficial. Eles podem ser operados por organizações governamentais ou não-governamentais e têm o direito de acessar a função eID. Os terminais de autenticação estrangeiros sempre exigem o eID PIN para vincular a leitura de dados com o consentimento do usuário e ao mesmo tempo vincular o cartão ao usuário. Eles são usados, por exemplo, para autenticação online. Os terminais de autenticação domésticos têm permissão de usar o CAN como senha do PACE. O uso do CAN não permite a verificação que o usuário do cartão e o titular do cartão são a mesma pessoa, de forma que essa verificação deve ser realizada através do identificação do usuário através da foto. Esses terminais são usados quando certos dados e funcionalidades do cartão precisam ser alterados depois da personalização do cartão.

- Terminais de assinatura confirmada: para geração de assinatura eletrônica qualificada.

- Terminais não autenticados: para certas operações administrativas executadas

localmente pelo titular do cartão, a autenticação do chip ou do terminal não é necessária.

Terminal type		**PACE password**	**Possible terminal rights**
Inspection system (official domestic/official foreign)	*General Authentication Procedure*	CAN; MRZ	• Read access to DG1 (MRZ), DG2 (facial image) of the biometric application and data of the eID application • Read access to DG3 (fingerprints) of the biometric application if access right proven by the authentication procedure
	eRP only: *Standard ePassport Inspection Procedure*	CAN; MRZ	Read access to DG1 (MRZ), DG2 (facial image) of the biometric application
	eRP only: *Advanced ePassport Inspection Procedure*	CAN; MRZ	Read access to DG1 (MRZ), DG2 (facial image), DG3 (fingerprints) of the biometric application
Authentication terminal (official domestic or non-official/foreign)		eID-PIN; CAN if *CAN allowed* right is proven	Read/write access to the data groups of the eID application according to authenticated rights Special rights: • Generation of a signature key pair • Setting the eID PIN, activating/deactivating eID application • Pseudonym • Age verification • Community ID verification
Confirmed Signature Terminal		CAN	• Generation of qualified signatures with additional entry of the signature PIN • Setting a new signature PIN with additional entry of the old signature PIN
		eID-PIN	• Creating the signature PIN • Terminating the key for qualified signatures and of the signature PIN
Unauthenticated terminal		eID-PIN	Setting a new eID PIN
		PUK	Resetting the retry counters of eID PIN/signature PIN

Figura 5.3 (tirada de [24]) – Tipos de terminais suportados pelo nPA.

5.1. Acesso aos Dados do Cartão

Os dados armazenados no cartão só podem ser acessados depois de uma autenticação do terminal bem sucedida através dos protocolos PACE, TA e CA. A Figura 5.4 ilustra o Procedimento de Autenticação Genérico usado pelo nPA.

Chip	Terminal
Reading the file EF.CardAccess	
	Entering/reading the PACE password (eID PIN/CAN/MRZ)
PACE (section 3.1.1)	
Transmitting the certificate chain Terminal Authentication (section 3.1.2)	
Reading the file EF.CardSecurity	
	Passive Authentication EF.CardSecurity (section 3.1.3)
Chip Authentication (section 3.1.4)	
Authentication terminal (optional): Document validity query (section 4.4.1) *Authentication terminal* (optional): Reading the revocation token (section 4.4.2)	
	Authentication terminal (optional): Revocation list query – only possible if the card is still valid (section 5.3)
Inspection system: reading of the EF.SOD	
	Inspection system: Checking the signature of the EF.SOD file (Passive Authentication)
Optional: Reading the approved data (section 3.2.2), Exercising the special rights (section 4.4.3)	
	Inspection system: Comparing the hash values of the data groups read to the values stored in the EF.SOD file

Figura 5.4 (tirada de [24]) – Procedimento de Autenticação Genérico.

Certos dados do sistema necessários para lidar com os protocolos de acesso, assim como senhas para o PACE, são armazenados no chip em um Arquivo Mestre (Master File - MF). Os arquivos do Arquivo Mestre são ilustrados na Figura 5.5. Os arquivos EF.CardAccess, EF.CardSecurity e EF.ChipSecurity, armazenados no Arquivo Mestre, contêm as informações de segurança mostradas na Figura 5.6.

File	Contents	Access Right		
		Read	Write	Internal Use
EF.ATR	According to [CEN 15480] part 2, contains Minimal Card Capabilities Descriptor (CCD) according [CEN 15480] part 3	always	-	-
EF.DIR	List of card applications ([CEN 15480] part 2)	always	-	-
EF.CardAccess	See section 3.2.4.	always	-	-
EF.CardSecurity	See section 3.2.4.	PACE+TA2	-	-
EF.ChipSecurity	See section 3.2.4.	PACE+TA2 as IS or AT with right *Privileged Terminal*	-	-
	MRZ password	-	-	For PACE
	CAN	-	-	For PACE
	eID PIN	-	PACE with eID PIN; AT + *PIN management*	For PACE
	PUK	-	-	For PACE
	Trust points for Terminal Authentication	Returned by PACE	When importing a link certificate	-
	Private key for Chip Authentication whose public key is included in EF.CardSecurity.	-	-	For CA after PACE + TA2
	Private key for Chip Authentication whose public key is included in EF.ChipSecurity.	-	-	For CA after PACE + TA2 as IS or AT with right *Privileged Terminal*
AT: Authenticated *Authentication Terminal* (PACE with eID-PIN or CAN (with *CAN allowed* right), TA2, CA2);				

Figura 5.5 – Arquivos no Arquivo Mestre.

- EF.CardAccess
  - PACEInfo
  - ChipAuthenticationInfo
  - ChipAuthenticationDomainParameterInfo
  - PrivilegedTerminalInfo
  - TerminalAuthenticationInfo
  - CardInfoLocator
- EF.CardSecurity
  - PACEInfo
  - ChipAuthenticationInfo
  - ChipAuthenticationDomainParameterInfo
  - ChipAuthenticationPublicKeyInfo
  - TerminalAuthenticationInfo
  - CardInfoLocator
  - RestrictedIdentificationInfo
  - RestrictedIdentificationDomainParameterInfo

  as well as the signature of this data, including the corresponding DS certificate.
- EF.ChipSecurity
  - PACEInfo
  - ChipAuthenticationInfo
  - ChipAuthenticationDomainParameterInfo
  - ChipAuthenticationPublicKeyInfo
  - PrivilegedTerminalInfo
  - TerminalAuthenticationInfo
  - CardInfoLocator
  - RestrictedIdentificationInfo
  - RestrictedIdentificationDomainParameterInfo
  - EIDSecurityInfo with hash values of data groups DG4, DG5, DG8 and DG9 of the eID application

  as well as the signature of this data, including the corresponding DS certificate.

Figura 5.6 – Informações de segurança dos arquivos EF.CardAccess, EF.CardSecurity, EF.ChipSecurity.

5.2. Função ePass/Aplicação biométrica

A função ePass (também chamada de aplicação biométrica) permite que o nPA seja usado como um passaporte eletrônico dentro da União Europeia. O acesso à aplicação biométrica do nPA é concedido unicamente a Sistemas de Inspeção autenticados. Os grupos de dados definidos pelo ICAO [2], listados na Tabela 5.1, são armazenados nessa aplicação. Os outros grupos de dados definidos pela ICAO não são usados, exceto o DG14 no caso da permissão de residência. Acesso a todos os grupos de dados (cartões eID) e ao DG3 (permissão de residência) é condicionado à prova dos direitos correspondentes através da autenticação do terminal.

File	Contents	Read access right
EF.COM (eRP only)	List of the available data groups and version information according to [ICAO 9303] (the use of this data group is not recommended because this data group is not signed.)	eRP only: BIS/EIS
EF.SOD	Hash values of the data groups DG1, DG2, DG3; signature over these hash values and the DS certificate (according to [ICAO 9303])	IS; eRP only: BIS/EIS
EF.CVCA (eRP only)	Trust points for certificate chain of the *inspection system* role of Terminal Authentication	eRP only: BIS/EIS
DG1	Data of the machine readable zone (MRZ) as printed on the card body	IS; eRP only: BIS/EIS
DG2	Digital facial image, identical to the printed image	IS; eRP only: BIS/EIS
DG3	Two fingerprints (optional in the ID card, mandatory in the residence permit). If no fingerprints are stored, this data group contains a random value.	IS + *Read DG3;* eRP only: EIS + *Read DG3*

DG14 (eRP only)	Contains the following `SecurityInfo` ([TR-03110]):
	`ChipAuthenticationInfo`
	`ChipAuthenticationPublicKeyInfo`
	`TerminalAuthenticationInfo`
	The public key contained for chip authentication is identical to the key from EF.ChipSecurity.
		eRP only: BIS/EIS

IS: Authenticated *Inspection System* (PACE with CAN or MRZ, TA2, CA2); BIS: *Basic Inspection System* (BAC/PACE); CA1 if supported by the terminal; EIS: *Extended Inspection System* (BAC/PACE; CA1; TA1)		

Tabela 5.1 (tirada de [24]) – Arquivos da Aplicação Biométrica

5.3. Função eID

A função/aplicação eID do cartão é uma função opcional que permite que o titular do cartão se identifique e autentique tanto a uma terceira pessoa como a serviços de e-government ou e-business. A Tabela 5.2 mostra os grupos de dados da aplicação eID. Esses dados não são assinados para prevenir que provedores de serviços os repassem para terceiros com prova criptográfica de autenticidade. Ao invés disso, a integridade e autenticidade dos dados são implicitamente asseguradas através do canal cifrado criado durante o protocolo CA. Para acessar dados do chip, os terminais tem que primeiro provar direito de acesso através de um certificado eletrônico de autorização.

Autenticação Online: para usar a função eID em transações online, o usuário precisa dos seguintes componentes:

- cartão eID, ou nPA, com a função eID ativada;
- leitora de cartão local;
- aplicação eID-Client (software AusweisApp). Essa aplicação implementa protocolos de segurança e interage com o usuário.

O servidor de serviços precisa de:

- um portal de serviços offline ou online com um certificado de autorização que concede ao provedor de serviços um acesso controlado aos dados do cartão;
- um eID-Server que processe a autenticação comunicando-se com o nPA usando protocolos criptográficos.

A infraestrutura para a realização da função eID, e o processo de comunicação dentro dessa infraestrutura são ilustrados na Figura 5.7.

File	Contents	Access Right		
		Read	Write	Internal use
DG1	Document type	IS; AT + *Read DG1*	-	-
DG2	Issuing country ("D" for Germany)	IS; AT + *Read DG2*	-	-
DG3	Expiration date format: YYYYMMDD	IS; AT + *Read DG3*	-	AT
DG4	First name(s)	IS; AT + *Read DG4*	-	-
DG5	Family name	IS; AT + *Read DG5*	-	-
DG6	ePA: Religious order/artistic name eRP: not used	IS; AT + *Read DG6*	-	-
DG7	Doctoral degree	IS; AT + *Read DG7*	-	-
DG8	Date of birth format: YYYYMMDD	IS; AT + *Read DG8*	-	-
DG9	Place of birth as unformatted text	IS; AT + *Read DG9*	-	-
DG10 - DG16	Not used	-	-	-
DG17	Address	IS; AT + *Read DG17*	AT + *Write DG17*	-
DG18	Community ID	IS; AT + *Read DG18*	AT + *Write DG18*	AT + *Community ID Verification*
DG19	eRP: Auxiliary Conditions I ePA: not used	IS; AT + *Read DG19*	AT + *Write DG19*	-
DG20	eRP: Auxiliary Conditions II ePA: not used	IS; AT + *Read DG20*	AT + *Write DG20*	-
DG21	Not used	-	-	-
	Reference date of birth for age verification	-	-	AT + *Age Verification*
	Key for service-provider-specific revocation token (section 4.4.2)	-	-	AT
	Key for service and card-specific identification (section 4.4.3.3)	-	-	AT + *Restricted Identification*

IS: Authenticated *Inspection System* (PACE with CAN or MRZ, TA2, CA2);
AT: Authenticated *Authentication Terminal* (PACE with eID-PIN or CAN (with *CAN allowed*), TA2, CA2);

Tabela 5.2 (tirada de [24]) – Arquivos da Aplicação *eID*.

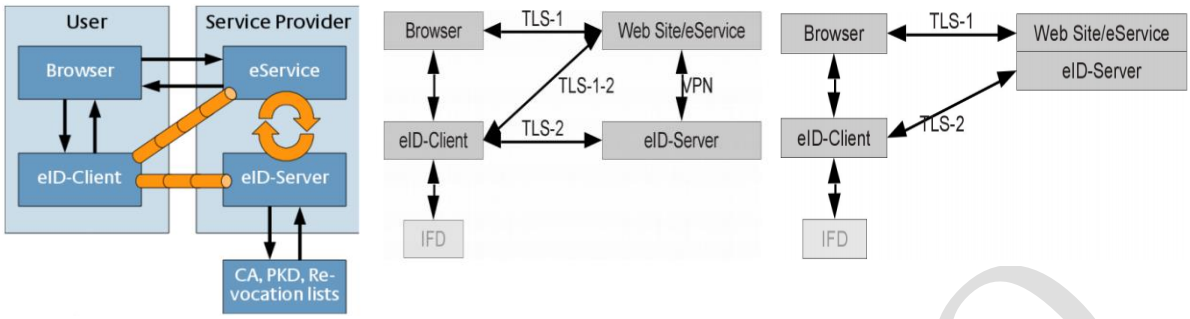

Figura 5.7: Infraestrutura da função *eID* (tiradas de [21], [22], [22] respectivamente da esquerda para direita).

O diagrama da esquerda na Figura 5.7 mostra a comunicação entre o usuário e o provedor de serviços.

O provedor de serviços (Service Provider) compreende o eService e um eID-Server. Um eID-Server (Servidor-eID) deve ser implementado para facilitar o uso da função eID com eServices (isto é, a autenticação online). O eID-Server fornece uma Interface-eID para encapsular a complexidade da função eID para eServices.

O eID-Server estabelece uma comunicação com a implementação do lado do cliente (eID-Cliente) e acessa certificados de autorização do terminal, listas de revogação e certificados CSCA da infraestrutura de chave pública (PKI). O eID-Server pode ser implementado como um servidor logicamente autônomo tal que vários eServices possam usá-lo e também pode ser operado por terceiros. Os dados trocados entre a Interface-eID são sempre assinados e devem ser cifrados se enviados numa rede aberta para proteger a confidencialidade, autenticidade e integridade dos dados processados. A Interface-eID é compartilhada entre o eID-Server e o eService.

O eService implementa a aplicação lógica da aplicação web apresentada no browser do usuário. O eService usa a eID-Interface do eID-Server para oferecer autenticação online mútua para seus usuários.

No lado do usuário temos o browser e um software chamado de eID-Client, executado no computador do usuário. O software usado pelo nPA é o AusweisApp.

A aplicação eID-Client implementa: a) um PIN-pad para leitoras de cartão sem entrada segura de PIN (isto é, leitora da categoria Cat-B); b) a exibição de informações do certificado CVCA do eService e restrições de direitos de acesso do eService; c) estabelecimento de canal seguro e vinculação de canal para autenticação online, e d)

(opcional) pré-verificação do certificado do eService antes de exibi-lo.

O diagrama do meio na Figura 5.7 ilustra o processo de comunicação entre o browser do usuário, o software eID-Client, o e-Service, e o eID-Server.

A autenticação online começa com um canal TLS, TLS-1, já existente entre o browser do usuário e o e-Service. A autenticação online é realizada entre o eID-Client e o eID-Server usando um segundo canal TLS, TLS-2. Em geral, o TLS-1 e o TLS-2 terminam em domínios diferentes. Um canal TLS intermediário, TLS-1-2, entre o eID-Client e o eService é necessário para permitir a vinculação entre o TLS-1 e o TLS-2. O eService e o eID-Server devem se comunicar usando um canal cifrado e íntegro mutualmente autenticado. Se o eService e o eID-Server se comunicam através de uma rede aberta (por exemplo, Internet), eles devem se comunicar usando uma comunicação segura por TLS. No caso especial em que ambos os canais TLS-1 e TLS-2 terminam no mesmo domínio, o canal intermediário TLS-1-2 não é necessário. Isso é ilustrado no diagrama da direita na Figura 5.7. Deve-se ressaltar que o eService deve usar o mesmo certificado TLS para os canais de comunicação com o browser do usuário e com a eID-Client.

O procedimento para autenticação online é mostrado na Figura 5.8, e descrito abaixo.

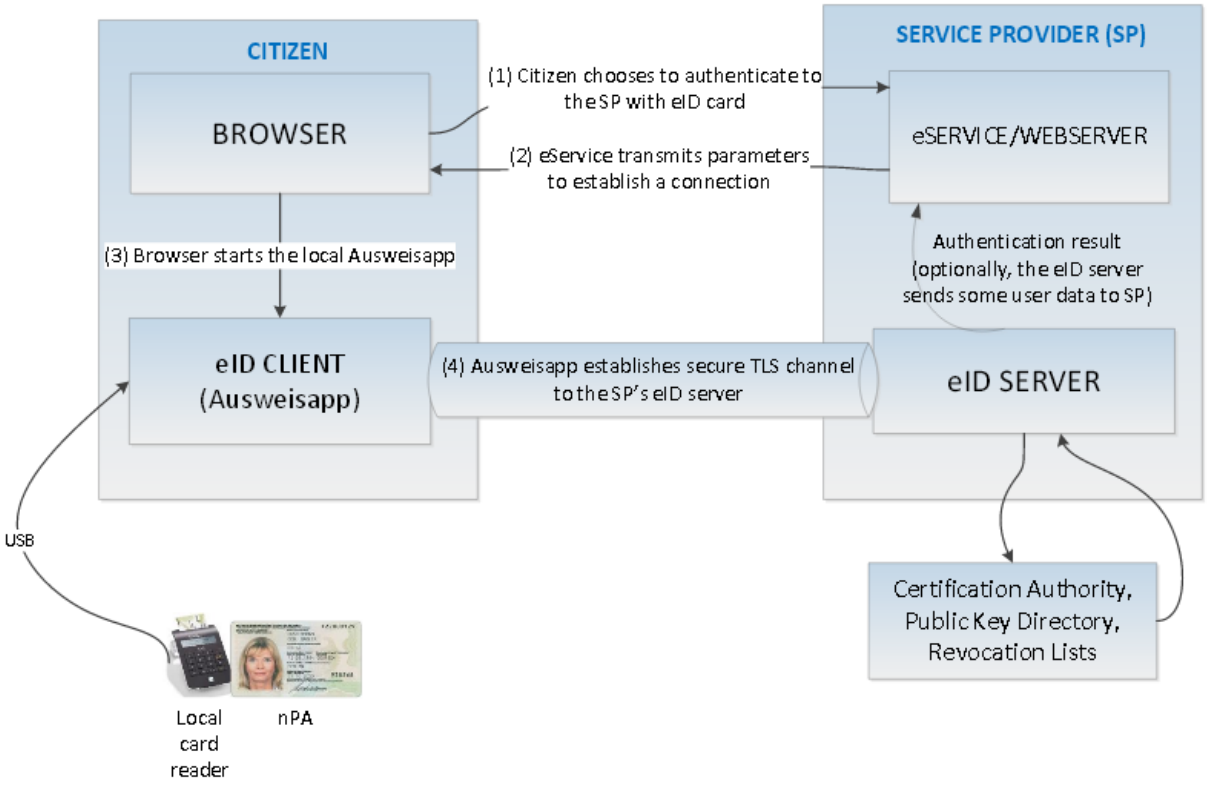

Figura 5.8 [tirada de [19]]: Procedimento de autenticação *online* do nPA.

O usuário visita um site online e solicita um serviço do provedor de serviços. O provedor de serviços requer que o usuário se autentique com o cartão de identidade eletrônico.

1. O eService envia uma webpage com um link embutido e todos os parâmetros necessários para o browser do usuário de forma a estabelecer uma conexão segura entre o eID-Client e o eID-Server. O link instrui o eID-Client a puxar a informação de endereço necessária e o certificado X.509 do servidor TLS, que será comparada com o CVC (Card Verifiable Certificate – Certificado Verificado por Cartão) do eID-Server.

2. O usuário clica no link e o browser do usuário executa um comando HTTP GET para inicializar a aplicação eID-Client (Ausweisapp).

3. A aplicação eID-Client estabelece um canal TLS com o eID-Server e exibe os dados do usuário requisitados pelo provedor de serviços para o usuário no seu browser. O cidadão decide que dados o Provedor de Serviços terá permissão de obter. Através

da inserção do PIN, o cidadão dá o seu consentimento para que os dados sejam acessados e transmitidos. Outro canal de troca segura de mensagens é estabelecido por cima do canal TLS pelas chaves geradas durante os protocolos de Controle de Acesso Estendido (EAC); dentro desse canal seguro e confiável a autenticação ocorre e os dados ou atributos do usuário requisitados são enviados para o eID-Server.

4. O eID-Server verifica os atributos do usuário e executa a sua autenticação. Ele transmite o resultado da autenticação para o provedor de serviços e, opcionalmente, envia os atributos do usuário se requisitados pelo provedor de serviços; contanto, os atributos só são enviados se o provedor de serviços tem o direito de ler esses atributos e a transmissão for autorizada pelo usuário. Em uma autenticação bem sucedida, o provedor de serviços concede acesso ao serviço ao usuário.

A Figura 5.9 ilustra a sequência dos protocolos criptográficos durante a autenticação online entre o nPA e um provedor de serviços. A autenticação online é um caso especial de acesso por um terminal de autenticação com o Procedimento de Autenticação Genérico visto na Seção 5.1.

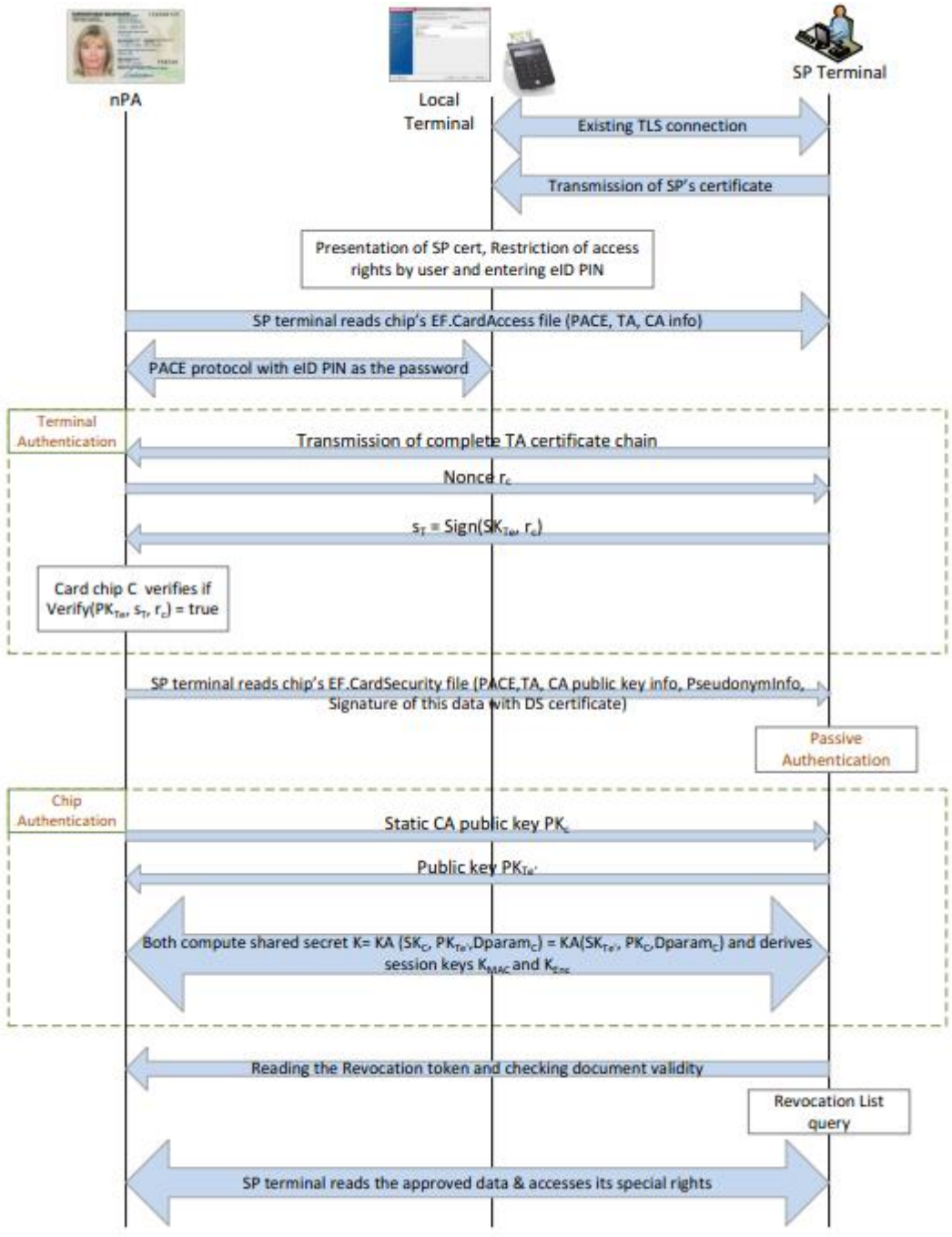

Figura 5.9 (tirada de [19]): Sequência de execução dos protocolos de autenticação para função *eID* do nPA.

A Figura 5.9 ilustra a comunicação entre o nPA e o terminal local, ou seja, a leitora do cartão e o terminal do provedor de serviços durante o processo de autenticação online. Os passos dessa comunicação são descritos abaixo.

1. Primeiramente a comunicação online precisa que já exista uma conexão TLS entre o terminal local e o terminal do provedor de serviços.

2. Quando o usuário requisita um serviço ao provedor de serviços, o provedor de serviços envia o seu certificado para o usuário, incluindo os seus direitos de acesso.

3. O certificado do servidor de serviços juntamente com os dados requisitados para acesso serão apresentados ao usuário pelo software Ausweisapp ou no display de uma leitora da categoria Cat-K.

4. O usuário antes de inserir o PIN eID para confirmação, pode restringir os dados que serão acessados pelo provedor de serviços.

5. O provedor de serviços lê o arquivo EF.CardAccess (ver Figura 5.6) armazenado no chip do cartão. Observe que esse arquivo já pode ser lido pelo provedor de serviços pois o arquivo é sempre legível por qualquer terminal. O EF.CardAccess contém informações como PACEInfo, CAInfo, e TAInfo.

6. O usuário insere o PIN eID e o protocolo PACE é executado entre o chip do cartão e o terminal local, ou leitora do cartão. O PACE autentica o usuário com a leitora do cartão como o dono legítimo do cartão através da inserção do PIN; ele também cria um canal seguro entre o cartão e a leitora para prevenir que o cartão seja lido a distância sem ter recebido acesso pelo dono do cartão. A partir daí o chip do cartão e a leitora do cartão iniciam a troca segura de mensagens usando o protocolo Secure Messaging com a chave de sessão criada no protocolo PACE.

7. O protocolo TA é iniciado entre o terminal do provedor de serviços e o chip. A comunicação entre o terminal do provedor de serviços e a leitora do cartão é feita usando o protocolo TLS, e a comunicação entre o chip e a leitora de cartão, usando o protocolo Secure Messaging com a chave de sessão PACE.

8. Uso do protocolo PA: após a execução do protocolo TA, o terminal lê o arquivo EF.CardSecurity (ver Figura 5.6) e compara o seu conteúdo com as informações lidas do arquivo EF.CardAccess antes da execução do PACE para verificar se as informações não foram modificadas. O terminal então verifica a assinatura dos conteúdos protegidos do EF.CardSecurity, já que ele conhece a chave pública do CSCA, e verifica a validade do certificado do DS. Depois de uma verificação bem sucedida de ambos a assinatura e o certificado, o terminal se assegura da autenticidade e integridade dos dados no cartão.

9. Aqui a autenticidade do chip é verificada através do protocolo CA. A chave pública efêmera do terminal calculada pelo chip durante o CA é comparada com a chave pública efêmera gerada pelo terminal durante o TA. Se elas são iguais, então ambos o chip e o terminal vão concordar com uma chave compartilhada secreta. Então o chip deriva chaves de sessão baseado no chave compartilhada secreta e gera um token de autenticação que é verificado pelo terminal no próximo passo. O chip é autenticado se a verificação é feita com sucesso. Então o chip reinicia o protocolo Secure Messaging com as novas chaves de sessão geradas durante o protocolo CA.
10. O terminal do provedor de serviços verifica a lista de revogação e a validade do cartão.
11. O terminal do provedor de serviços lê os dados aprovados.

Vemos que as mensagens trocadas entre o chip e o terminal do provedor de serviços após a execução do protocolo PACE (durante a execução dos protocolos TA, PA, e CA) podem ser cifradas e decifradas pelo terminal local (leitora do cartão). Como parte do processo de Autenticação do Chip (protocolo CA), um canal seguro fim-a-fim é estabelecido entre o chip e o terminal do provedor de serviços para que o terminal local só repasse qualquer comunicação entre o chip e o terminal do provedor de serviços.

Autenticação Offline: a grande diferença entre a autenticação online e a autenticação offline usando o nPA é que:

- na autenticação off-line, o usuário é autenticado com o terminal do provedor de serviços pela iniciação do PACE quando ele insere o PIN. Dessa forma, um canal seguro de comunicação é estabelecido entre o cartão e o terminal do provedor de serviços pelas chaves de sessão PACE;

- na autenticação online, o PACE é usado para compartilhamento do PIN e para criar um canal seguro somente entre o cartão e a leitora de cartão, ou terminal local. O canal TLS estabelecido pelo software Ausweisapp com o eID-Server do provedor de serviços protege a comunicação entre o eID-Client (a aplicação Ausweisapp que é conectada com a leitora do cartão e o cartão) e o eID-Server até que as chaves de sessão do protocolo CA sejam estabelecidas.

5.3.1. Funções de Minimização de Dados

Para proteger a privacidade dos usuários e reduzir casos de fraude online (tais como roubo de identidade), o nPA oferece funções de minimização de dados. Essas funções incluem identificação restrita com uso de pseudônimos, verificação de idade, e verificação de residência. O objetivo é transmitir para o provedor de serviços uma quantidade mínima de dados que são necessários somente para uma transação específica.

5.3.1.1. Identificação Restrita e o uso de Pseudônimos

O nPA permite que o dono do cartão se identifique e autentique a um provedor de serviços usando um pseudônimo. Isso é feito através do protocolo de Identificação Restrita (RI – Restricted Identification) proposto pelo BSI [5], e é destinado especialmente para o uso em autenticação na Internet. O protocolo RI é opcional e supostamente executado apenas depois que um canal seguro com a aplicação ou serviço requisitado tiver sido construído através do protocolo Secure Messaging. Note que mesmo que o chip tenha sido autenticado com o provedor de serviços através de uma chave pública certificada (usando o protocolo CA), esse chave ainda é compartilhada com um grande grupo de chips (para permitir anonimato através de domínios diferentes); assim o chip não é unicamente identificável. Por esse motivo, a Identificação Restrita introduz pseudônimos gerados a partir da identidade do dono do cartão e da identificação pública do provedor de serviços. Para ser usado em autenticação online esses pseudônimos tem que satisfazer as seguintes propriedades:

- domain-specific linkability: essa propriedade requer que terminais de provedores de serviços dentro de um mesmo domínio possam reconhecer cartões de identidade unicamente sem conseguir qualquer informação sobre o titular do cartão. Portanto, o pseudônimo de um certo chip pode se autenticar unicamente com um terminal e esse terminal pode traçar perfis de usuários;

- cross-domain anonymity: essa propriedade requer que pseudônimos só sejam vinculados dentre de um mesmo domínios. Dois provedores de serviços associados a domínios diferentes não devem conseguir vincular transações com um mesmo usuário.

O protocolo RI funciona da seguinte forma.

1. O terminal envia seu identificador público PK_sector e certos parâmetros do domínio dparam para o chip. Esses parâmetros são diferentes dos parâmetros de domínio usados nos protocolos PACE e EAC.
2. O chip verifica a validade de PK_sector usando a chave pública da autoridade certificadora PK_CA que foi usada como parte do certificado do terminal transmitido durante o protocolo TA do EAC.
3. O chip calcula o pseudônimo de domínio específico **DS_nym= Hash(DH((PK_sector,dparam),SK_ID))**,onde SK_ID é uma chave secreta única do chip. O chip então envia DS_nym para o terminal.
4. O terminal verifica se o pseudônimo DS_nym está numa lista negra e caso contrário o chip é autenticado.

Note que o chip não conhece o identificador público PK_ID correspondente a sua chave secreta única SK_ID; somente SK_ID é armazenada no chip. Da mesma forma, o terminal não sabe a chave secreta correspondente ao seu identificador de domínio público. Essas chaves são usadas externamente somente para gerar listas de revogação (listas negras).

O protocolo RI deve ser executado usando Secure Messaging depois do protocolo CA. Isso é necessário para que o terminal tenha a garantia de que o chip, e consequentemente o pseudônimo é confiável. Como o protocolo TA é executado antes do protocolo RI, o identificador público do terminal, PK_sector, também é confiável.

5.3.1.2. Verificação de Idade

Ao invés de transmitir a data de nascimento, o nPA permite que o chip envie para o terminal somente a informação se o titular do cartão tem mais de X anos ou não. Para isso, o terminal deve ser autorizado a usar essa função. Uma data de teste será enviada para o chip como parte do protocolo TA. Após ser solicitado a responder a pergunta “O titular do

cartão tem mais de X anos?", o chip irá verificar se a data de nascimento do titular do cartão é posterior a data de teste e responderá simplesmente com "sim" ou "não", ao invés de revelar a data de nascimento armazenada.

5.3.1.3. Verificação de Residência

O terminal pode verificar se o titular do cartão mora em determinada região. O chip irá comparar o identificador regional armazenado com um identificador regional de referência enviado pelo terminal durante o protocolo TA (como parte dos dados auxiliares). Essa função pode ser usada para oferecer serviços localizados para cidadãos que residem em determinada área.

.

5.4. Função eSign

Para que o titular do nPA possa usar a função eSign, ele precisa de um PIN especial para assinatura eletrônica, o PIN eSign, juntamente com o PIN eID. Para assinar um documento usando assinatura eletrônica, o usuário deve ter um certificado de assinatura online de uma Autoridade Certificadora (CA – Certification Authority). O processo funciona da seguinte forma.

1. Geração de chaves: um par de chaves pública e privada é gerado dentro do cartão. A chave privada sk_sign, é mantida em segredo dentro do chip, e a chave pública, pk_sign, é pública.

2. Obtenção de certificado para pk_sign: um certificado de assinatura online para pk_sign é obtido de um provedor de serviços aprovado para certificação (uma CA); o titular da eID se identifica com a CA com o eID PIN e solicita o certificado enviando pk_sign. O certificado é então carregado no nPA.

3. Assinatura de documentos online: para assinar documentos online, o usuário deve colocar o nPA na leitora e inserir PIN eSign. A assinatura gerada é uma assinatura eletrônica qualificada (QES), que pela lei alemã, é equivalente a uma assinatura feita à mão. A assinatura eletrônica, portanto, constitui uma prova que pode ser mostrada no tribunal ou em qualquer procedimento administrativo se for necessário.

5.5. Revogação de Cartões e Certificados de Terminais

Como vimos anteriormente, vários cartões nPA compartilham a mesma chave pública para autenticação do chip, pk_CA. Isso se torna um problema se um cartão nPA é perdido ou roubado, e um atacante consegue obter a chave secreta de autenticação do chip, sk_CA. Note que essa é a única informação sobre o cartão trocada durante o processo de autenticação, e portanto que pode ser usada para revogação. Assim, se uma chave sk_CA for extraída, um grande número de cartões nPA terão que ser bloqueados. Esse número pode chegar a alguns milhões, já que todos os chips produzidos dentro de um período de 3 meses tem o mesmo par de chaves de autenticação do chip. Isso quer dizer que a segurança inteira do sistema de identidades eletrônicas alemãs é baseada na resistência do chip contra adulterações.

Caso um nPA seja perdido ou roubado, o titular deve ser capaz de revogar o cartão. Assim, um serviço de revogação de cartões nPA deve estar disponível 24 horas por dia durante os 7 dias da semana mesmo quando o titular do cartão estiver fora. A princípio, isso iria precisar do armazenamento de todas as informações pessoais necessárias para a identificação do titular do cartão. Contudo, a lei alemã preza bastante a privacidade de seus cidadãos e não permite a geração de tal base central de dados dos cidadãos. Para contornar esse problema, a base de dados do serviço de revogação armazena códigos de revogação únicos para cada titular de cartão. O código de revogação é o hash da concatenação da data de nascimento do titular do cartão, do seu sobrenome, do seu nome e de uma senha de revogação recebida através de uma carta. Isso garante a privacidade e permite a revogação de cartões nPA sem a necessidade de um registro central com informações pessoais dos cidadãos.

Além de o nPA poder ser revogado se o cartão for perdido ou roubado, ele também pode ser revogado somente para um serviço específico. Para isso, o nPA usa listas de revogação específicas para o serviço. Isso significa que durante o processo de identificação eletrônica, o cartão transmite um atributo de revogação específico para o serviço e para o cartão, que o provedor de serviços compara com sua lista de revogação específica do serviço.

Para a revogação de certificados de terminais, é necessário que tais certificados

tenham uma validade curta, isto é, de alguns dias. O problema é que o chip não contém um relógio interno porque o cartão não possui fonte de energia própria; como ele não sabe a data atual, não tem como saber se o certificado foi revogado. O chip sabe apenas que a data atual é posterior a data de emissão de certos certificados que foram recebidos, tais como o C_CVCA, C_DV, e certificados de sistemas de inspeção ou terminais de autenticação oficiais. Para permitir que o titular do cartão atualize a data aproximada do seu cartão, a data pode ser atualizada através de um certificado atualizado diariamente. O certificado deve ser do terminal do tipo autenticação doméstico oficial, disponível online.

5.6. Ataques contra o nPA

- Uso de um keylogger: keylogger é um tipo de software que permite o registro de informações digitadas em um teclado. Em agosto e setembro de 2010, o Chaos Computer Club (CCC) alegou que leitoras da categoria Cat-B não são seguras; como leitoras dessa categoria não possuem um PIN pad, o usuário deve inserir o PIN através do teclado de um computador. No cenário de ataque, o CCC assume que um atacante pode instalar um keylogger no computador do usuário sem que ele note. O PIN do usuário é obtido quando o mesmo tenta se autenticar com a leitora do cartão através do inserção do PIN. No entanto, o atacante só pode se autenticar ilicitamente se o dono do cartão esquecer de tirar o cartão da leitora. Além disso, esse ataque pode ser facilmente evitado se o usuário do cartão mantiver o computador protegido contra vírus.

- Ataque do tipo man-in-the-middle (MitM): esse tipo de ataque está relacionado com a segurança dos canais de comunicação TLS entre o usuário e o provedor de serviços (ver Figura 6.7) e não com os mecanismos de autenticação usados pelo nPA.

REFERÊNCIAS

[1] ICAO Doc 9303, Machine Readable Travel Documents - Part 1: Machine Readable Passports, Volume 1: Passports with Machine Readable Data Stored in Optical Character Recognition Format, 6th Edition, 2006.

[2] ICAO Doc 9303, Machine Readable Travel Documents - Part 1: Machine Readable Passports, Volume 2: Specifications for Electronically Enabled Passports with Biometric Identification Capability, 6th Edition, 2006.

[3] ICAO Technical Report, Supplemental Access Control for Machine Readable Travel Documents, Version 1.1, April 2014.

[4] BSI TR-03110-1, Advanced Security Mechanisms for Machine Readable Travel Documents – Part 1: eMRTDs with BAC/PACEv2 and EACv1, Version 2.10, March 2012.

[5] BSI TR-03110-2, Advanced Security Mechanisms for Machine Readable Travel Documents – Part 2: Extended Access Control Version 2 (EACv2), Password Authenticated Connection Establishment (PACE), and Restricted Identification (RI), Version 2.10, March 2012.

[6] BSI TR-03110-3, Advanced Security Mechanisms for Machine Readable Travel Documents – Part 3: Common Specifications, Version 2.10, March 2012.

[7] www.bsi.bund.de

[8] Y. Liu, T. Kasper, K. Lemke-Rust, C. Paar: E-passport: *Cracking basic access control keys with Copacabana*. In: SHARKS 2007.

[9] J. Bender, D. Kügler: *Introducing the PACE solution; Countries need to be aware of the need to replace BAC*. In: Keesing Journal of Documents and Identity, issue 30, 2009.

[10] M. Ullmann, M.Vögeler: *Contactless Security Token Enhanced Security by Using New Hardware Features in Cryptographic-Based Security Mechanisms*. In: Towards Hardware-Intrinsic Security 2010: 259--279.

[11] G. Avoine, K. Kalach, J. Quisquater: ePassport: Securing international contacts with contactless chips. In: Financial Cryptography 2008: 141—155.

[12] B. Deufel, C. Mueller, G. Duffy, T. Kevenaar: *BioPACE – Biometric passwords for next generation authentication protocols for machine-readable travel documents.* Datenschutz und Datensicherheit – DuD, 37(6): 363—366, 2013.

[13] N. Buchmann, R. Peeters, H. Baier, A. Pashalidis: *Security considerations on extending PACE to a biometric-based connection establishment.* BIOSIG 2014: 15—26.

[14] FRONTEX: Operational and technical security of electronic passports. Warsaw, July 2011.

http://frontex.europa.eu/assets/Publications/Research/Operational_and_Technical_Security_of_Electronic_Pasports.pdf

[15] Heise Online. CCC zeigt Sicherheitsprobleme beim elektronischen Personalausweis auf [Update], September 2010. http://www.heise.de/newsticker/meldung/CCC-zeigt-Sicherheitsprobleme-beim-elektronischen-Personalausweis-auf-Update-1083649.html

[16] Heise Online. Elektronischer Personalausweis Sicherheitsdefizite bei Lesegeräten [Update], August 2010. http://www.heise.de/newsticker/meldung/Elektronischer-Personalausweis-Sicherheitsdefizite-bei-Lesegeraeten-Update-1064338.html

[17] Heise Online. The H Security. CCC reveals security problems with German electronic IDs, September, 2010. http://www.h-online.com/security/news/item/CCC-reveals-security-problems-with-German-electronic-IDs-1094577.html

[18] Ö. Dagdelen: *The cryptographic security of the german electronic identity card.* PhD thesis, Technische Universität Darmstadt, Alemanha, 2013.

[19] B. B. Hampiholi: *Secure & privacy-preserving eID systems with attribute-based credentials*. MSc thesis, Universiteit Twente, Holanda, 2014.

[20] BSI TR-03110-3, Advanced Security Mechanisms for Machine Readable Travel Documents – Part 3: Common Specifications, Version 2.11, July 2013.

[21] BSI TR-03130-1, Technical Guideline eID-Server – Part 1: Functional Specification, Version 2.01, January 2014.

[22] BSI TR-03124-1, Technical Guideline TR-03124-1 eID-Client – Part 1: Specifications, Version 1.1, May 2014.

[23] FUTURE ID, Survey and Analysis of Existing eID and Credential Systems, April 2013.

[24] BSI TR-03127, Technical Guideline TR-03127 Architecture electronic identity card and electronic residence permit, Version 1.13, March 2011.

[25] BSI TR-03119, Technical Guideline TR-03119 Requirements for Smart Card Readers Supporting eID and eSign Based on Extended Access Control, Version 1. 3, March 2013.

[26] http://www.feig.de

Universidade de Brasília – UnB

Centro de Apoio ao Desenvolvimento Tecnológico – CDT

Laboratório de Tecnologias da Tomada de Decisão – LATITUDE

www.unb.br – www.cdt.unb.br – www.latitude.eng.br