Análise Formal de Requisitos de Consistência para Atribuições de Valoração

Frederico P. Santiago[1], **Mário S. Alvim**[2]

[1] Universidade Federal de Minas Gerais, Departamento de Engenharia Elétrica
frede-stg@ufmg.br

[2]Universidade Federal de Minas Gerais, Departamento de Ciência da Computação
msalvim@dcc.ufmg.br

Abstract. *Quantitative information flow aims to assess and control the leakage of sensitive information by computer systems. The community has recently turned its attention to semantic-based measures of information, which employ worth assignments to attribute a different worth to each portion of the secret, depending on its sensitivity. Algorithms for automatically deriving worth-assignments from a scenario of interest must be guided by a set of consistency requirements, which are expensive to compute and enforce. In this paper we analyze several candidates for consistency requirements, identifying subsets of redundant requirements and reducing, hence, the computational effort required by the algorithm for automatically deriving appropriate worth assignments.*

Resumo. *O campo de fluxo de informação quantitativo (QIF, do inglês* quantitative information flow*) concerne à mensuração da informação sigilosa que vaza por meio de sistemas computacionais. A comunidade recentemente tem se focado em métricas de informação baseadas em semântica, que empregam atribuições de valoração para atribuir a cada porção do segredo uma valoração correspondente à sua importância. Algoritmos para derivar atribuições de valoração a partir de um cenário de interesse devem ser guiados por um conjunto de requisitos de consistência, que são caros de se computarem. Neste artigo analisamos vários candidatos a requisitos de consistência, identificando subconjuntos de requisitos redundantes e reduzindo, assim, o esforço computacional para o algoritmo de derivação automática de atribuições de valoração.*

1. Introdução

O campo de *fluxo de informação quantitativo* (QIF, do inglês *quantitative information flow*) concerne à mensuração da informação sigilosa que vaza, para um adversário não-autorizado, por meio de sistemas computacionais. Assume-se que o adversário possui *informação a priori* sobre os segredos antes de a execução do sistema ser iniciada—modelada como uma distribuição de probabilidades sobre os segredos—, e que ele tem acesso aos *efeitos observáveis* do sistema em execução. Combinando a informação *a priori* e os observáveis públicos, o adversário obtém *informação a posteriori* sobre os segredos—modelada como uma coleção de distribuições de probabilidades sobre segredos condicionadas às observações feitas. O *vazamento de informação* de uma execução é computado como a diferença entre a informação *a posteriori* e a informação *a priori*.

Esta definição de vazamento depende de como informação é medida, e diferentes *métricas de informação* quantificam diferentes aspectos de interesse do cenário analisado.

Métricas populares no campo de QIF incluem *entropia de Shannon* [Clark et al. 2005, Malacaria 2007, Moskowitz et al. 2003, Chatzikokolakis et al. 2008, Alvim et al. 2012a], que mede o número esperado de tentativas que um adversário precisa para identificar o segredo usando pesquisa binária; *guessing entropy* [Massey 1994, Malacaria 2015], que mede o número esperado de tentativas que um adversário precisa para identificar o segredo usando pesquisa linear; e *probabilidade de acerto* [Smith 2009, Braun et al. 2009], que mede a probabilidade de o segredo ser inferido em um certo número de tentativas.

Todas estas métricas, entretanto, possuem a limitação de considerar todos os segredos como atômicos e de igual valor, e recentemente a comunidade tem considerado cenários mais ricos. O arcabouço de *g-leakage* [Alvim et al. 2012b, McIver et al. 2014, Alvim et al. 2014a, Alvim et al. 2016] utiliza funções de ganho para quantificar o benefício de diferentes prognósticos do adversário sobre o segredo. Escolhendo uma função de ganho adequada é possível modelar a importância de cada porção do segredo. Entretanto, identificar funções de ganho suficientemente expressivas, porém não complexas demais, é uma tarefa desafiadora. Em [Alvim et al. 2014b] são identificadas várias dimensões utilizadas na interpretação operacional de métricas clássicas de informação—o *tipo* e a *quantidade* de questões que o adversário pode utilizar, assim como a sua *probabilidade de acerto*—e propõem a inclusão de uma nova dimensão: a *valoração* de cada parte do segredo. Os autores consideram que um segredo possa ser particionado em *campos*, que podem ser combinados para formar *estruturas*. Como o vazamento de estruturas distintas corresponde a ameaças distintas, uma *atribuição de valoração* é utilizada para associar uma *valoração* a cada estrutura. Por exemplo, o segredo correspondendo à conta bancária de um cliente pode conter duas estruturas de 8 dígitos: sua senha de internet e seu número de telefone. Como o vazamento da senha tem o potencial de causar um dano considerável, a estrutura correspondente teria uma valoração alta; como o número de telefone é público, a estrutura correspondente teria uma valoração baixa.

Usando atribuições de valoração, Alvim et al. generalizaram métricas tradicionais para refletir a semântica dos segredos, e provaram diversas propriedades destas novas métricas. Entretanto, tais generalizações assumem a existência uma atribuição de valoração adequada para cada cenário de interesse. A fim de sintetizar semi-automaticamente atribuições de valoração a partir de um cenário de interesse, Alvim et al. propuseram um algoritmo iterativo que recebe como entrada informações básicas sobre o cenário de interesse—i.e., a distribuição de probabilidaade a priori sobre os segredos e quais estruturas são consideradas intrinsecamente sensíveis—e deriva a valoração das demais estruturas condicionadas a um conjunto de *requisitos de consistência*, que são propriedades matemáticas que garantem o bom comportamento das atribuições de valoração. Vários requisitos de consistência podem ser considerados, como, por exemplo, a não-negatividade, a monotonicidade, e a propriedade de que se uma estrutura "carrega informação sobre uma outra", então sua valoração deve refletir esta "informação compartilhada". O algoritmo de Alvim et al., entretanto, é indiferente aos requisitos de consistência adotados, e simplesmente tenta propagar a valoração entre estruturas garantindo um requisito por vez, em sequência, o que pode levar a dois problemas durante a execução do algoritmo: (i) se os requisitos adotados não forem consistentes entre si, o algoritmo pode divergir; e (ii) mesmo que o conjunto de requisitos seja consistente, se ele for redundante, o algoritmo se torna desnecessariamente lento por testar individualmente requisitos que já são implicados por outros requisitos já verificados.

Neste trabalho realizamos uma análise formal de requisitos de consistência para atribuições de valoração que pode guiar o projetista do sistema a escolher um conjunto de requisitos auto-consistentes e, a partir deste conjunto, decidir se existe uma ordem de verificação entre estes requisitos que torne o algoritmo de síntese de atribuições de valoração mais eficiente, ou se os requisitos escolhidos devem ser necessariamente checados individualmente. Mais especificamente, nossas principais contribuições são:

1. Uma análise da relevância de diversas propriedades matemáticas candidatas a requisitos de consistência, incluindo: não-negatividade, monotonicidade, princípio da inclusão-exclusão, princípio da correlação e separabilidade estatística;
2. Uma análise da auto-consistência entre tais requisitos. Em particular, demonstramos que a negatividade e a correlação produzem sempre distribuições triviais incompatíveis com o requisito de separabilidade estatística.
3. Uma análise da redundância entre conjuntos de requisitos, identificando se a satisfabilidade de um conjunto de requisitos implica necessariamente a satisfabilidade de outros requisitos. Em particular, demonstramos que à parte de casos triviais: correlação implica não-negatividade e monotonicidade; correlação e monotonicidade implicam não-negatividade; e correlação e não-negatividade implicam monotonicidade.
4. Relacionado ao item acima, uma análise de quais requisitos de consistência não permitem otimização na checagem. Em particular, demonstramos que correlação deve ser checada independentemente de monotonicidade, não-negatividade e separabilidade estatística.

O restante deste artigo está organizado da seguinte forma. A Seção 2 apresenta os conceitos fundamentais de fluxo de informação quantitativo e o arcabouço de Alvim et al. de métricas baseadas em atribuições de valoração. A Seção 3 formaliza candidatos a requisitos de consistência e analisa a relevância prática de cada um, enquanto na Seção 4 investiga a relação entre conjuntos de requisitos, identificando subconjuntos inconsistentes e subconjuntos redundantes. Por fim, a Seção 5 discute trabalhos futuros e conclui.

2. Preliminares

Nesta seção revisamos os fundamentos do campo de fluxo de informação quantitativo e o arcabouço de Alvim et al. [Alvim et al. 2014b] de métricas baseadas em atribuições de valoração.

2.1. Segredos, estruturas e atribuições de valoração

Seguindo [Alvim et al. 2014b], consideramos que cada segredo pode ser decomposto em partes atômicas chamadas de *campos*. O conjunto finito desses campos é dado por $\mathcal{F} = \{f_1, f_2, \ldots, f_m\}$, e o domínio de cada campo é o conjunto de todos os possíveis valores que o campo pode assumir, denotado por $domain(f_i)$, $1 \leq i \leq m$. Uma *estrutura* é um subconjunto $\mathfrak{f} \subseteq \mathcal{F}$. Se $\mathfrak{f} = \{f_{i_1}, f_{i_2}, \ldots, f_{i_k}\}$ então $domain(\mathfrak{f}) = domain(f_{i_1}) \times \cdots \times domain(f_{i_k})$. O conjunto de todas as possíveis estruturas é o conjunto potência $\mathcal{P}(\mathcal{F})$ dos campos. Chamamos $\mathfrak{f}$ de *estrutura máxima* quando $\mathfrak{f} = \mathcal{F}$.

Um *segredo* s é um mapeamento da estrutura máxima para valores na forma $s = \langle s[f_1], \ldots, s[f_m] \rangle$, onde $s[f_i] \in domain(f_i)$ é o valor assumido pelo campo f_i.

Desta forma, o conjunto $\mathcal{S}$ de todos os possíveis segredos é igual a $domain(\mathcal{F})$. Quando fazemos a projeção de um segredo s em uma estrutura $\mathfrak{f}$, obtemos o *subsegredo* $s[\mathfrak{f}]$. O conjunto de todos os subsegredos associados a uma estrutura $\mathfrak{f}$ é $\mathcal{S}[\mathfrak{f}] = domain(\mathfrak{f})$.

Exemplo 1. Considere um sistema bancário em que o segredo é composto pelos campos $\mathcal{F} = \{c, s, d\}$, representando, respectivamente, o número da conta c, a senha s da conta, e a data de nascimento d de cada correntista. O domínio $domain(c)$ de contas é o conjunto de todos os identificadores de contas, o domínio $domain(s)$ de senhas é o conjunto de todos os números de 4 dígitos, e o domínio $domain(d)$ de datas de nascimento são todos os dias de $01/01/1\,900$ até hoje. Como o conjunto de estruturas é o conjunto potência $\mathcal{P}(\mathcal{F})$ do conjunto de campos, podemos representar todas as estruturas em um reticulado em que a relação de ordem parcial é derivada da relação de inclusão em conjuntos. A Figura 1 representa o reticulado correspondente ao exemplo bancário. Note que a estrutura $\{c, s\}$ (conta e senha do correntista) está incluída na estrutura $\{c, s, d\}$ (a informação completa sobre o correntista), mas ela não tem relação de ordem com a estrutura $\{d\}$ (a data de nascimento do correntista). Considere o segredo $s = \langle \texttt{CONTA-ABC}, 2510, 11/06/1996\rangle$, representando um correntista nascido em 11/06/1996, dono da conta "CONTA-ABC", e tendo como senha "2510". Podemos isolar o subsegredo $s[\{c, s\}] = \langle \texttt{CONTA-ABC}, 2510\rangle$ correspondente à sua conta e senha, e o subsegredo $s[d] = \langle 11/06/1996\rangle$ correspondente à sua data de nascimento. □

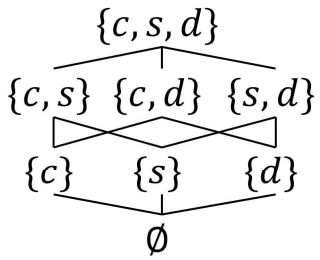

Figura 1. Reticulado de estruturas do Exemplo 1.

As estruturas podem ter valores associados com a informação que carregam. Uma *atribuição de valoração* é uma função $\omega : \mathcal{P}(\mathcal{F}) \to \mathbb{R}$ do conjunto de todas as estruturas para os reais. A valoração pode ser interpretada como o ganho do adversário ao tomar conhecimento de uma estrutura ou como o dano que o protetor do segredo sofre caso o adversário tome conhecimento desta estrutura. Por exemplo, no Exemplo 1 para refletir o fato de que a senha de um correntista é mais sensível que sua data de nascimento, podemos atribuir uma valoração tal que $\omega(\{s\}) > \omega(\{d\})$.

A atribuição de valoração deve representar apropriadamente a sensibilidade da estrutura no cenário em que está inserida, levando em consideração três aspectos relevantes: (i) o quanto o protetor do segredo se preocupa em proteger o conteúdo da estrutura em questão; (ii) qual é o ganho do adversário em descobrir esta estrutura; e (iii) o quanto esta estrutura revela a respeito de outras. Adotando uma atribuição de valoração apropriada, também é possível modelar outros aspectos de interesse, tais como: *vazamento baseado em semântica* (o impacto do vazamento é um reflexo da qualidade da informação vazada, e não do número de bits vazados); *compartilhamento de segredos* ou *secret sharing* (estruturas que isoladamente carregam pouca informação podem carregar muita informação se reveladas em conjunto); e *correlação de estruturas* (se uma estrutura carrega informação sobre outra, sua valoração deve refletir isso).

2.2. Métricas de informação baseadas em atribuições de valoração

Usando atribuições de valoração é possível generalizar métricas clássicas, como entropia de Shannon, *guessing entropy* e probabilidade de acerto para considerar segredos não-

atômicos e seus aspectos semânticos. [Alvim et al. 2014b] classifica as métricas com base em valoração em três categorias:

- *W-Measures* medem a valoração extraída do sistema em um ataque com número fixo de tentativas e uma determinada probabilidade de sucesso do adversário que deve ser atingida. E.g., *worth of certainty*, *W-vulnerability* e *worth of expectation*.
- *P-Measures* medem a probabilidade de sucesso do adversário em um ataque com um valor determinado de valoração extraída e número de tentativas fixo. E.g., *W-probability of guessing*.
- *N-Measures* medem quantas tentativas são necessárias para que o adversário extraia uma certa quantidade de valoração do sistema e tenha uma determinada probabilidade de sucesso. E.g., *W-guessing entropy* e *W-Shannon entropy*.

2.3. Um algoritmo para sintetizar atribuições de valoração

Todas as métricas baseadas em atribuição de valoração assumem que exista uma atribuição de valoração que capture os aspectos relevantes do cenário de interesse. [Alvim et al. 2014b] propõe um algoritmo para a sintetização de atribuições de valoração a partir de três aspectos relevantes do cenário de interesse: o conhecimento do adversário, requisitos de segredo e os requisitos de consistência.

- **Conhecimento do adversário** é o conjunto de informações relevantes ao adversário sobre o sistema que será atacado. Essas informações vêm de fontes externas e constituem o conhecimento *a priori* do adversário. A configuração do conhecimento do adversário é modelada como uma distribuição de probabilidades sobre as estruturas.
- **Requisitos de segredo** transmitem as preocupações do protetor do segredo em relação às estruturas que o compõem. Toda estrutura pode ser classificada como *intrinsecamente sensível*, isto é, possui valor por si só, ou como *contingencialmente sensível*, que possui valor apenas pela informação que revela sobre as intrinsecamente sensíveis. Os requisitos de segredo mapeiam todas as estruturas intrinsecamente sensíveis a uma valoração considerada apropriada pelo projetista do sistema. A relação entre as valorações de estruturas intrinsecamente sensíveis norteia o algoritmo na atribuição de valoração a todas as outras de maneira consistente com os interesses do projetista. No Exemplo 1 a senha do cliente deve ser considerada intrinsecamente sensível, já que é informação chave para o acesso à conta. Já a data de nascimento do correntista não é imprescindível para o acesso, mas sabe-se que existe uma parcela razoável de usuários de sistemas protegidos por senha que utilizam suas datas de nascimento como senha, o que faz da data de nascimento uma estrutura contingencialmente sensível.
- **Requisitos de consistência** são propriedades consideradas essenciais para garantir o significado operacional da valoração e para guiar o algoritmo na atribuição das valorações pelo reticulado.

Provido com as entradas acima, o algoritmo proposto segue os seguintes passos.

1. Construir o reticulado de estruturas completo.
2. Usar os requisitos de segredo para atribuir as valorações adequadas para as estruturas intrinsecamente sensíveis.

3. Derivar a distribuição de probabilidade p_S. Pode-se considerar que as partições do reticulado são variáveis aleatórias e usar p_S para encontrar as distribuições de probabilidades das estruturas.
4. Considerar uma métrica ν bem estabelecida e consistente com o cenário em questão e, para toda estrutura do reticulado, atualizar sua valoração respeitando os requisitos de consistência. Repetir até que todas as estruturas possuam valorações consistentes com os requisitos de consistência impostos.

O ponto crítico do algoritmo acima é o passo 4, em que se verificam iterativamente os requisitos de consistência. E.g., para testar se toda a distribuição é não negativa, basta uma comparação por elemento de $\mathcal{P}(\mathcal{F})$, mas para testar correlação deve-se garantir que cada elemento está devidamente relacionado a todos os outros do reticulado. O que pode se tornar uma atividade bastante custosa computacionalmente. Nas próximas seções vamos reduzir o número de requisitos de consistência para os quais devemos testar as distribuições sem abdicar das propriedades de interesse.

3. Os requisitos de consistência

Nesta seção formalizamos e avaliamos a significância prática de várias propriedades matemáticas candidatas a serem requisitos de consistência.

Os requisitos de consistência impostos à atribuição da valoração podem ser variados, contanto que sejam de interesse de quem constrói o sistema e que não haja conflitos entre as propriedades desejadas. [Alvim et al. 2014b] propõe algumas propriedades possivelmente relevantes. São elas: não-negatividade, monotonicidade, inclusão-exclusão e separabilidade estatística (no artigo original, chamado de independência). Além destes, consideraremos o princípio da correlação em nossos estudos.

- **Não-negatividade (NNG).** Nenhuma estrutura possui valoração negativa,

$$\forall \mathfrak{f} \in \mathcal{P}(\mathcal{F}) : \omega(\mathfrak{f}) \geq 0.$$

- **Monotonicidade (MNT).** Uma estrutura deve carregar, no mínimo, tanta informação quanto qualquer uma de suas subestruturas.

$$\forall \mathfrak{f}, \mathfrak{f}' \in \mathcal{P}(\mathcal{F}) : \mathfrak{f} \subseteq \mathfrak{f}' \Rightarrow \omega(\mathfrak{f}') \geq \omega(\mathfrak{f}).$$

- **Separabilidade Estatística (SEP).** Se duas estruturas $\mathfrak{f}$ e $\mathfrak{f}'$ forem *estatisticamente independentes* (denotado por $\mathfrak{f} \perp \mathfrak{f}'$), isto é, não compartilham informação entre si, então:

$$\omega(\mathfrak{f}, \mathfrak{f}') = \omega(\mathfrak{f}) + \omega(\mathfrak{f}').$$

- **Correlação (COR).** A correlação condiciona a valoração de uma estrutura à quantidade de informação que ela fornece a respeito de outras estruturas. A métrica de correlação que utilizaremos neste estudo é a *vulnerabilidade condicional*. A vulnerabilidade $V(\mathfrak{f})$ nos fornece a probabilidade de acerto de uma estrutura $\mathfrak{f}$ em apenas uma tentativa. Formalmente,

$$V(\mathfrak{f}) = \max_{x \in domain(\mathfrak{f})} p(s[\mathfrak{f}] = x).$$

A vulnerabilidade condicional $V(\mathfrak{f}' \mid \mathfrak{f})$, portanto, nos fornece a probabilidade de acerto da estrutura $\mathfrak{f}'$ dado que sabemos o conteúdo da estrutura $\mathfrak{f}$. Formalmente,

$$V(\mathfrak{f}' \mid \mathfrak{f} = x) = \max_{x' \in domain(\mathfrak{f}')} p(s[\mathfrak{f}'] = x' \mid \mathfrak{f} = x). \quad (1)$$

A propriedade da correlação é dada por

$$\forall \mathfrak{f}, \mathfrak{f}' \in \mathcal{P}(\mathcal{F}) : \omega(\mathfrak{f}) \geq V(\mathfrak{f}' \mid \mathfrak{f}) \cdot \omega(\mathfrak{f}').$$

Esta propriedade é interessante por creditar às estruturas suas influências nas valorações de todo o reticulado.

- **Inclusão-Exclusão (IE).** Utilizado largamente em outras métricas, o princípio da inclusão-exclusão proposto é da forma:

$$\forall \mathfrak{f}, \mathfrak{f}' \in \mathcal{P}(\mathcal{F}) : \omega(\mathfrak{f} \cup \mathfrak{f}') = \omega(\mathfrak{f}) + \omega(\mathfrak{f}') - \omega(\mathfrak{f} \cap \mathfrak{f}').$$

Percebeu-se, porém, que esta não é uma propriedade desejável no sistema, visto que sua imposição contraria propriedades mais elementares. E.g., sejam o número da conta bancária de um indivíduo e sua senha os dois campos que constituem um segredo. É razoável condicionar que o conhecimento da conta e da senha tenha uma valoração consideravelmente maior que a soma das valorações das partes. Mas pela definição da inclusão-exclusão, a maior valoração que a junção de duas estruturas pode atingir é a soma das valorações de cada estrutura.

4. As relações entre os requisitos

Nesta seção reportamos os principais resultados do nosso estudo: as relações de auto-consistência e redundância identificadas entre requisitos de consistência. Em particular, mostramos que para todo caso não trivial, o requisito de correlação implica não-negatividade e monotonicidade. Mostramos, também, que a separabilidade estatística não implica nenhuma das outras propriedades e que a correlação não é derivável das demais consideradas.

Começaremos definindo funções distribuição de valoração triviais.

Definição 2. *Uma atribuição de valoração $\omega : \mathcal{P}(\mathcal{F}) \to \mathbb{R}$ é trivial (TRIV) se, e somente se* $\forall \mathfrak{f}, \mathfrak{f}' \in \mathcal{P}(\mathcal{F}) : \omega(\mathfrak{f}) = \omega(\mathfrak{f}')$.

Nas demonstrações que se seguem, faremos referência constante ao cenário-base apresentado no exemplo abaixo..

Exemplo 3. Considere o cenário em que haja dois campos $\mathcal{F} = \{f_1, f_2\}$, gerando o conjunto de estruturas $\mathcal{P}(\mathcal{F}) = \{\emptyset, \{f_1\}, \{f_2\}, \{f_1, f_2\}\}$; Por simplicidade, vamos chamar $\mathfrak{f} = \{f_1\}$, $\mathfrak{f}' = \{f_2\}$ e $\mathfrak{f}'' = \{f_1, f_2\}$. A Figura 2 representa o reticulado de estruturas para este exemplo. Assuma que os campos são binários, i.e., que $domain(f_1) = domain(f_2) = \{0, 1\}$, de forma que o conjunto de todos os segredos possíveis seja $\mathcal{S} = \{00, 01, 10, 11\}$. A Tabela 1(a) apresenta uma distribuição de probabilidades sobre os

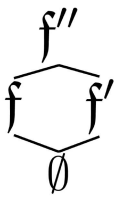

Figura 2. Reticulado das estruturas do Exemplo 3.

Tabela 1. Campos estatisticamente independentes para o Exemplo 3.

(a) Distribuição de probabilidades do segredo.

$s[f_1]$	$s[f_2]$	p_s
0	0	$1/6$
0	1	$1/3$
1	0	$1/6$
1	1	$1/3$

(b) Vulnerabilidades condicionais $V(a \mid b)$.

$b \backslash a$	$\emptyset$	$\mathfrak{f}$	$\mathfrak{f}'$	$\mathfrak{f}''$
$\emptyset$	1	$1/2$	$1/2$	$1/3$
$\mathfrak{f}$	1	1	$2/3$	$2/3$
$\mathfrak{f}'$	1	$2/3$	1	$2/3$
$\mathfrak{f}''$	1	1	1	1

segredos, e a Tabela 1(b) apresenta as vulnerabilidades condicionais (veja Equação (1)) entre cada par de campos. Note que neste exemplo temos que os campos são estatisticamente independentes ($f_1 \perp f_2$). □

Também precisaremos do seguinte resultado auxiliar.

Lema 4. *Se uma atribuição de valoração ω satisfaz o requisito de correlação, então não pode haver duas estruturas com valoração de sinais diferentes. Formalmente, temos que COR $\Rightarrow \forall \mathfrak{f}, \mathfrak{f}' \in \mathcal{P}(\mathcal{F}) : \omega(\mathfrak{f}) \cdot \omega(\mathfrak{f}') \geq 0$.*

Demonstração. Considere estruturas $\mathfrak{f}_1, \ldots, \mathfrak{f}_n$ e $\mathfrak{f}$ de um segredo tais que $\mathfrak{f}_1 \cup \ldots \cup \mathfrak{f}_n \subseteq \mathfrak{f}$ (Note que fazendo $n \geq 1$, temos que sempre podemos encontrar tal conjunto de estruturas). Se a atribuição de valoração ω satisfizer COR, então para $i = 1, \ldots, n$ temos que $\omega(\mathfrak{f}_i) \geq V(\mathfrak{f} \mid \mathfrak{f}_i) \cdot \omega(\mathfrak{f})$ e que $\omega(\mathfrak{f}) \geq V(\mathfrak{f}_i \mid \mathfrak{f}) \cdot \omega(\mathfrak{f}_i)$. Lembrando que a vulnerabilidade condicional $V(a \mid b)$ é sempre não-negativa, note que há dois casos a se analisar para o sinal de $\mathfrak{f}$: se $\omega(\mathfrak{f}) \geq 0$, por COR temos que $\omega(\mathfrak{f}_i) \geq 0$; e se $\omega(\mathfrak{f}) < 0$, por COR temos que $\omega(\mathfrak{f}_i) \leq \omega(\mathfrak{f}) < 0$. Isso implica que todas as estruturas $\mathfrak{f}_1, \ldots, \mathfrak{f}_n$ devem ter o mesmo sinal que a estrutura $\mathfrak{f}$. Escolhendo $\mathfrak{f}$ como a estrutura máxima do reticulado, que contém todas as demais, segue que todas as estruturas de um reticulado devem possuir sinal igual à estrutura máxima. □

Agora estamos prontos para enunciar nossos resultados principais.

Proposição 5 ($\neg$TRIV $\wedge$ COR $\Rightarrow$ NNG)**.** *Se uma atribuição de valoração ω não-trivial satisfaz o requisito de correlação, então ω necessariamente também satisfaz o requisito de não-negatividade.*

Demonstração. Vamos demonstrar a proposição equivalente de que se uma atribuição de valoração ω satisfizer a propriedade de correlação e possuir ao menos uma estrutura de valor negativo, então ω deve necessariamente ser trivial (i.e., COR $\wedge$ $\neg$NNG $\Rightarrow$ TRIV).

Considere um reticulado que contenha ao menos uma estrutura de valoração negativa. Pelo Lema 4, todas as demais estruturas devem também possuir valorações negativas. Assumindo a correlação, $\forall \mathfrak{f}, \mathfrak{f}' \in \mathcal{P}(\mathcal{F})$ temos que $V(\mathfrak{f}' \mid \mathfrak{f}) \cdot \omega(\mathfrak{f}') \leq \omega(\mathfrak{f}) < 0$ e que $V(\mathfrak{f} \mid \mathfrak{f}') \cdot \omega(\mathfrak{f}) \leq \omega(\mathfrak{f}') < 0$. Como $\omega(\mathfrak{f})$ e $\omega(\mathfrak{f}')$ são negativos, é verdade que $\omega(\mathfrak{f}') \leq \omega(\mathfrak{f}) < 0$ e que $\omega(\mathfrak{f}) \leq \omega(\mathfrak{f}') < 0$. Mas note que como temos $\omega(\mathfrak{f}') \leq \omega(\mathfrak{f})$ e $\omega(\mathfrak{f}) \leq \omega(\mathfrak{f}')$, então $\omega(\mathfrak{f}) = \omega(\mathfrak{f}')$. Como a igualdade é válida para quaisquer $\mathfrak{f}$ e $\mathfrak{f}'$, a distribuição é trivial. □

Proposição 6. *(COR $\wedge$ NNG $\Rightarrow$ MNT) Se uma atribuição de valoração ω não-trivial satisfaz os requisitos de correlação e não-negatividade, então ω necessariamente também satisfaz monotonicidade.*

Demonstração. Se $\mathfrak{f}' \subseteq \mathfrak{f} \Rightarrow V(\mathfrak{f}' \mid \mathfrak{f}) = 1$. Lançando mão de COR e assumindo a NNG, temos que $\omega(\mathfrak{f}) \geq V(\mathfrak{f}' \mid \mathfrak{f}) \cdot \omega(\mathfrak{f}') = \omega(\mathfrak{f}')$, que é exatamente a definição de monotonicidade.

Note que a condição de não-negatividade é essencial, pois o fator $V(\mathfrak{f}' \mid \mathfrak{f})$ reduz $\omega(\mathfrak{f}')$ em módulo. Se $\omega(\mathfrak{f}') < 0$, a redução de seu módulo significa em aumento de seu valor de forma que $V(\mathfrak{f}' \mid \mathfrak{f}) \cdot \omega(\mathfrak{f}') \geq \omega(\mathfrak{f}')$. □

Proposição 7. *(MNT∧NNG∧SEP ⇏ COR) A propriedade da correlação não é derivável de monotonicidade, não-negatividade ou separabilidade estatística, e nem de nenhuma combinação entre estas.*

Demonstração. Como contra exemplo, tome o cenário do Exemplo 3 e considere a atribuição de valoração ω tal que $\omega(\emptyset) = 0$, $\omega(\mathfrak{f}) = 5$, $\omega(\mathfrak{f}') = 5$ e $\omega(\mathfrak{f}'') = 10$. Note que NNG e MNT são satisfeitas em ω, assim como SEP (por construção, os campos f_1 e f_2 são independentes e ω satisfaz $\omega(\mathfrak{f}'') = 10 = 5 + 5 = \omega(\mathfrak{f}) + \omega(\mathfrak{f}')$). Entretanto, COR não é satisfeita: note que $\omega(\emptyset) < V(\mathfrak{f} \mid \emptyset) \cdot \omega(\mathfrak{f})$. □

Proposição 8. *(MNT∧NNG∧COR ⇏ SEP) A propriedade da separabilidade estatística não é derivável de monotonicidade, correlação ou não-negatividade e nem de combinações entre estas.*

Demonstração. Já sabemos pela Proposição 6 que COR ∧ NNG ⇒ MNT. Então basta demonstrar que NNG∧COR ⇏ SEP, o que faremos com o contra-exemplo a seguir. Tome novamente o cenário do Exemplo 3, e considere uma atribução de valoração ω tal que $\omega(\emptyset) = 4$, $\omega(\mathfrak{f}) = 7$, $\omega(\mathfrak{f}') = 5$ e $\omega(\mathfrak{f}'') = 10$. Note que COR, NNG (e, consequentemente, MNT) são satisfeitas, porém SEP é violada, pois $\omega(\mathfrak{f}'') \neq \omega(\mathfrak{f}) + \omega(\mathfrak{f}')$. □

Proposição 9. *(SEP ⇏ MNT ∨ NNG ∨ COR) A separabilidade estatística não implica monotonicidade, correlação nem não-negatividade.*

Demonstração. Como contra-exemplo, tome o cenário do Exemplo 3 e considere a atribução de valoração ω tal que $\omega(\emptyset) = -5$, $\omega(\mathfrak{f}) = -5$, $\omega(\mathfrak{f}') = -5$ e Perceba que SEP é satisfeita, pois $\omega(\mathfrak{f}'') = \omega(\mathfrak{f}) + \omega(\mathfrak{f}')$, mas a distribuição é negativa (NNG desrespeitada) e não-trivial, o que implica, pela Proposição 5, que COR não é satisfeita. Além disto, $\omega(\mathfrak{f}'') < \omega(\mathfrak{f})$. Ou seja, a monotonicidade também é desrespeitada. □

Neste ponto observamos que apesar de apenas as Proposições 5 e 6 apresentarem resultados positivos (i.e., determinarem uma ordem de precedência entre requisitos), todas as proposições apresentadas são úteis para auxiliar o projetista do sistema na atribuição de valoração. Resultados negativos demonstram que certos conjuntos de requisitos são inconsistentes (e.g., a Proposição 5 demonstra que se houver ao menos uma valoração negativa na distribuição, assumindo correlação, esta distribuição deve ser trivial) ou que, mesmo se consistentes, alguns conjuntos de requisitos demandam que a checagem seja feita individualmente (e.g., a Proposição 7 demonstra que correlação deve ser checada independentemente de monotonicidade, não-negatividade e separabilidade estatística).

5. Conclusões e trabalhos futuros

Neste trabalho analisamos a relevância de vários requisitos de consistência para atribuições de valoração, e identificamos vários conjuntos de requisitos inconsistentes entre si ou redundantes.

Os resultados obtidos levam a dois pontos fundamentais na otimização do algoritmo de síntese de atribuições de valoração a partir de um cenário de interesse: Em primeiro lugar, é suficiente que testemos apenas COR, NNG e SEP para garantir a validade destas e de MNT, o que aumenta a eficiência do algoritmo. Em segundo lugar, não podemos abrir mão de NNG na distribuição sem gerar trivialidade, se considerarmos COR ou gerar inconsistência, se considerarmos COR e SEP.

Como trabalho corrente estamos investigando as consequências de assumir a independência dos campos na definição de separabilidade estatística, buscando maiores reduções no esforço computacional para este caso particular. Também propomos métricas que captam a relação entre as estruturas de maneiras diferentes como funções de correlação e estudamos suas consequências na atribuição de valoração. Por fim, pretendemos analisar quantitativamente a redução do esforço computacional atingida com a eliminação das redundâncias entre requisitos de consistência identificadas.

Referências

Alvim, M., Chatzikokolakis, K., McIver, A., Morgan, C., Palamidessi, C., and Smith, G. (2014a). Additive and multiplicative notions of leakage, and their capacities. In *Computer Security Foundations Symposium (CSF), 2014 IEEE 27th*, pages 308–322.

Alvim, M., Chatzikokolakis, K., McIver, A., Morgan, C., Palamidessi, C., and Smith, G. (2016). Axioms for information leakage. In *Computer Security Foundations Symposium (CSF), 2016 IEEE 29th*, pages 77–92.

Alvim, M. S., Andrés, M. E., and Palamidessi, C. (2012a). Quantitative information flow in interactive systems. *Journal of Computer Security*, 20(1):3–50.

Alvim, M. S., Chatzikokolakis, K., Palamidessi, C., and Smith, G. (2012b). Measuring information leakage using generalized gain functions. In *Proc. 25th IEEE Computer Security Foundations Symposium (CSF 2012)*, pages 265–279.

Alvim, M. S., Scedrov, A., and Schneider, F. B. (2014b). When not all bits are equal: Worth-based information flow. In *Proc. 3rd Conference on Principles of Security and Trust (POST 2014)*, pages 120–139.

Braun, C., Chatzikokolakis, K., and Palamidessi, C. (2009). Quantitative notions of leakage for one-try attacks. In *Proceedings of the 25th Conf. on Mathematical Foundations of Programming Semantics*, volume 249 of *Electronic Notes in Theoretical Computer Science*, pages 75–91. Elsevier B.V.

Chatzikokolakis, K., Palamidessi, C., and Panangaden, P. (2008). Anonymity protocols as noisy channels. *Inf. and Comp.*, 206(2–4):378–401.

Clark, D., Hunt, S., and Malacaria, P. (2005). Quantitative information flow, relations and polymorphic types. *J. of Logic and Computation*, 18(2):181–199.

Malacaria, P. (2007). Assessing security threats of looping constructs. In Hofmann, M. and Felleisen, M., editors, *Proceedings of the 34th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages, POPL 2007*, pages 225–235. ACM.

Malacaria, P. (2015). Algebraic foundations for quantitative information flow. *Mathematical Structures in Computer Science*, 25(2):404–428.

Massey (1994). Guessing and entropy. In *Proceedings of the IEEE International Symposium on Information Theory*, page 204. IEEE.

McIver, A., Morgan, C., Smith, G., Espinoza, B., and Meinicke, L. (2014). Abstract channels and their robust information-leakage ordering. In *Proc. 3rd Conference on Principles of Security and Trust (POST 2014)*, pages 83–102.

Moskowitz, I. S., Newman, R. E., Crepeau, D. P., and Miller, A. R. (2003). Covert channels and anonymizing networks. In *Proceedings of the 2003 ACM Workshop on Privacy in the Electronic Society*, WPES '03, pages 79–88, New York, NY, USA. ACM.

Smith, G. (2009). On the foundations of quantitative information flow. In de Alfaro, L., editor, *Proc. 12th International Conference on Foundations of Software Science and Computational Structures (FoSSaCS '09)*, volume 5504 of *Lecture Notes in Computer Science*, pages 288–302.

