

Cálculo de Ambientes Tipado para Aplicações Pervasivas Seguras

**Douglas Pereira Pasqualin¹, Juliana Kaizer Vizzotto¹,
Giovani Rubert Librelotto¹, André Rauber Du Bois²**

¹Programa de Pós-Graduação em Informática – Universidade Federal de Santa Maria
Avenida Roraima, 1000 – 91.105-900 – Santa Maria – RS

²Universidade Federal de Pelotas (UFPel)
Caixa Postal 354 – 96.001-970 – Pelotas – RS

{dpasqualin, juvizzotto, librelotto}@inf.ufsm.br, dubois@ufpel.edu.br

1. Introdução

A computação pervasiva preocupa-se em mudar a forma como interagimos com os computadores. Nesse novo paradigma, chamado de "computação para o século 21"[Weiser 1991], uma das principais características é a sensibilidade ao contexto, ou seja, as aplicações perceberem o ambiente e ao mesmo tempo se adaptarem automaticamente as necessidades do usuário.

Segundo [Campbell et al. 2003], a maioria das pesquisas em computação pervasiva foca na construção de infra-estruturas que irão permitir a interação entre dispositivos e redes sem fio ou no desenvolvimento de novas aplicações. Porém, segurança e privacidade não estão sendo devidamente exploradas.

Nesse sentido, o presente resumo propõe a definição de um sistema de tipos, que garanta segurança dos dados em aplicações, para o Cálculo de Ambientes [Cardelli and Gordon 1998]. O Cálculo de Ambientes pode ser visto como uma linguagem de programação básica para aplicações sensíveis ao contexto.

2. Sistema de tipos

Sistema de tipos é uma ferramenta utilizada em linguagens de programação que garante consistência de dados e permite detectar erros em programas antes da sua execução [Wright 2010]. Um tipo constitui a definição de um conjunto de valores, e as operações possíveis sobre esse tipo. Dessa forma, um sistema de tipos auxilia na detecção de erros, e previne comportamentos incorretos de programas.

Uma recente área de pesquisa em tipos é a de segurança de dados. Enquanto um sistema de tipos tradicional aplica regras associando valores a tipos de dados, um tipo de dados para segurança pode aplicar o mesmo princípio de semântica, determinando o proprietário de um determinado pedaço de informação. Essas anotações podem ajudar a garantir a integridade dos dados do sistema [Wright 2010].

Um trabalho proposto na literatura que utiliza sistema de tipos para desenvolver sensibilidade ao contexto é o trabalho proposto em [Dapoigny and Barlatier 2007].

3. Cálculo de Ambientes

Com o crescimento da *web* e mais especificamente dos dispositivos móveis, uma nova característica da computação emerge: a mobilidade de código. A grande dificuldade

de mover códigos através da *web* está no tratamento dos domínios administrativos [Cardelli and Gordon 1998]. Não basta somente conhecer o endereço IP do dispositivo de destino. *Firewalls* e políticas de segurança controlam os domínios, restringindo o que pode passar através dele. Dessa forma, mobilidade envolve também autorização para entrar ou sair de um certo domínio.

A abordagem utilizada pelos autores é estruturar hierarquicamente ambientes computacionais, confinando agentes a esses ambientes e movendo os ambientes através do controle desses agentes. A grande vantagem, segundo os autores, é que essa abordagem permite mover ambientes que podem estar contidos dentro dos ambientes a serem movidos, ao invés de mover somente objetos individualmente. Além disso, o Cálculo de Ambientes pode ser utilizado como base para novas linguagens de programação, tornando o código mais simples de entender e manter [Weis et al. 2006].

Uma abordagem que utiliza Cálculo de Ambientes com sistema de tipos para desenvolver sensibilidade ao contexto é o trabalho proposto em [Kjærgaard and Bunde-Pedersen 2006].

4. Considerações Finais

Através desse estudo inicial, é possível concluir que sistema de tipos é uma poderosa ferramenta que pode ser utilizada a fim de garantir a segurança e a consistência de programas. Assim, o foco deste trabalho de mestrado é propor uma linguagem de programação sensível ao contexto estaticamente tipada, tomando como base o Cálculo de Ambientes, visando principalmente à garantia da segurança dos dados de aplicações pervasivas.

Referências

- Campbell, R., Al-Muhtadi, J., Naldurg, P., Sampemane, G., and Mickunas, M. D. (2003). Towards security and privacy for pervasive computing. In *Proceedings of the 2002 Next-NSF-JSPS international conference on Software security: theories and systems*, ISSS'02, pages 1–15, Berlin, Heidelberg. Springer-Verlag.
- Cardelli, L. and Gordon, A. D. (1998). Mobile ambients. In *Proceedings of the First International Conference on Foundations of Software Science and Computation Structure*, pages 140–155, London, UK. Springer-Verlag.
- Dapoigny, R. and Barlatier, P. (2007). Towards a context theory for context-aware systems. In *Proceeding of the 2007 conference on Advances in Ambient Intelligence*, pages 36–55, Amsterdam, The Netherlands, The Netherlands. IOS Press.
- Kjærgaard, M. B. and Bunde-Pedersen, J. (2006). Towards a formal model of context awareness. In *Proceedings of the First International Workshop on Combining Theory and Systems Building in Pervasive Computing (Pervasive 2006)*, pages 667–674.
- Weis, T., Becker, C., and Brändle, A. (2006). Towards a programming paradigm for pervasive applications based on the ambient calculus. In *Proceedings of the First International Workshop on Combining Theory and Systems Building in Pervasive Computing (Pervasive 2006)*, Dublin, Ireland.
- Weiser, M. (1991). The computer for the 21st century. *Scientific American*, 265(3):94–104.
- Wright, A. (2010). Type theory comes of age. *Communications of the ACM*, 53(2):16–17.