# Um modelo de segurança para proteger arquivos pessoais

Evandro Fernandes Giovanini
Supervisor: Marco Dimas Gubitoso

## 1. Introdução

A segurança dos sistemas operacionais modernos foi criada pensando na realidade de décadas atrás, onde era comum que dezenas de usuários compartilhassem computadores, por exemplo em universidades. Nessa realidade o principal objetivo era proteger o sistema contra usuário maliciosos.

Nos dias de hoje é comum computadores serem pessoais, e ao invés de aplicativos serem instalados por um administrador de redes, o próprio usuário baixa e instala inúmeros aplicativos da internet, nem sempre de fontes confiáveis. O modelo de segurança do sistema operacional impede que um aplicativo executado por um usuário cause danos à instalação do sistema, mas não há nada que impeça o acesso completo a todos os arquivos pessoais do usuário. Assim, um aplicativo mal comportado ou mal intencionado pode, por exemplo, apagar arquivos, enviá-los pela rede sem autorização.

Nesse trabalho foi desenvolvido o SecApp, um arcabouço para contornar esse problema que permite rodar aplicativos no Linux de forma isolada, sem que eles tenham acesso irrestrito aos dados do usuário. Desta forma, mesmo que algum aplicativo tenha uma falha que pode potencialmente ser abusada, os dados pessoais continuam seguros e protegidos.

## 2. Funcionamento

O sistema operacional Linux segue uma estrutura padronizada de arquivos, onde os arquivos pessoais de cada usuário são armazenados no diretório *home*, e arquivos do sistema são armazenados em diretórios como *usr*, *var* e *etc*. O diretório que contém esses diretórios é a chamada raiz do sistema de arquivos.

O SecApp cria um ambiente especial de execução único para cada aplicativo. Neste ambiente os diretórios do sistema são replicados, para que os programas possam funcionar normalmente. A diferença está no diretório pessoal; ao invés de enxergar o diretório pessoal real, cada aplicativo terá um diretório específico. Usando como raiz esse ambiente especial o SecApp inicia um dado aplicativo. O aplicativo não enxerga além do ambiente em que está sendo executado, de modo que não irá ter acesso de leitura ou escrita aos arquivos pessoais do usuário.

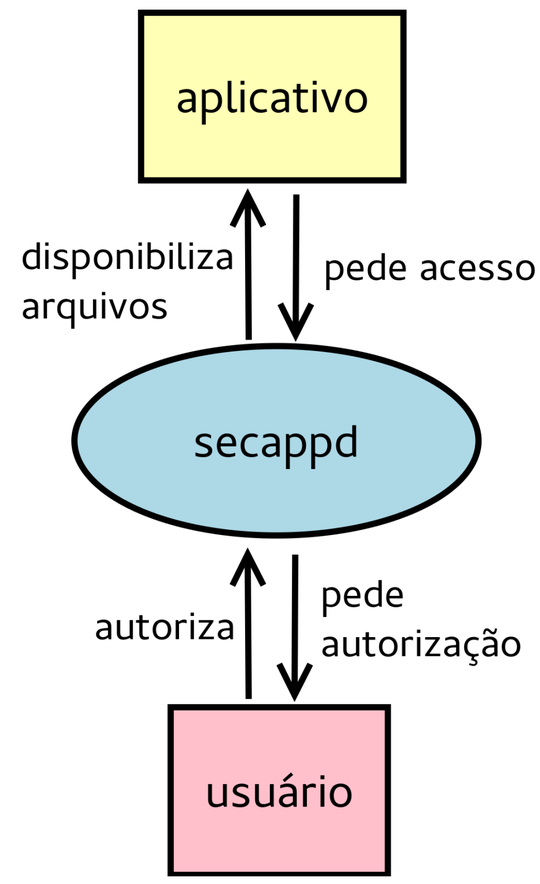
## 3. Concessão de acesso

O SecApp isola cada aplicativo em seu ambiente de execução, com seu próprio diretório de dados. Mas e quando o usuário deseja, por exemplo, enviar por e-mail uma planilha de cálculo ou documento de texto criado por outro aplicativo? De alguma forma é preciso que o programa de e-mail acesse arquivos criados por outros aplicativos.

Para essas situações o SecApp permite que aplicativos peçam permissão de leitura aos arquivos pessoais do usuário.

Com o SecApp, em toda sessão do usuário é executado um *daemon*, chamado *secappd*, com o qual aplicativos podem se comunicar via *DBus*. O *secappd* é executado no ambiente normal do usuário, e portanto com acesso a todos os seus arquivos pessoais.

Quando o *secappd* recebe uma requisição de concessão de acesso ele exibe um diálogo para o usuário. Caso o usuário aceite liberar o acesso a todos os arquivos pessoais, o *secappd* faz uma junção entre o diretório pessoal real e o diretório pessoal visto por aquele aplicativo. Desta forma, o aplicativo continua sua execução normalmente, mas agora passa a enxergar a estrutura real e completa do diretório pessoal. Esse processo, ilustrado abaixo, acontece imediatamente e de forma transparente ao aplicativo em execução.




## 4. Conclusão

O SecApp é uma forma simples de executar aplicativos com segurança, e já está pronto para ser utilizado em instalações do Linux. Veja mais informações em https://linux.ime.usp.br/~evandrog/secapp.