MBDA Italia

IRAD & Innovation

13 Marzo 2013

La Società MBDA

MBDA è stata creata nel 2001 a seguito dell'unione delle attività delle più importanti società europee nel settore: **Matra BAe Dynamics** in Francia e UK, **Aerospatiale Missiles** in Francia, le attività missilistiche dell'allora **Alenia Marconi Systems** in UK e Italia ed **EADS/LFK** in Germania. **MBDA** è controllata con stessi diritti societari da **BAE SYSTEMS** (37,5%), **EADS** (37,5%) e **FINMECCANICA** (25%), questo la rende la prima ed unica società Europea nel settore della Difesa pienamente integrata. Grazie alle sue riconosciute capacità e potenzialità **MBDA** può oggi puntare ancora di più al futuro. La società ha una spiccata connotazione verso l'Innovazione e l'eccellenza tecnologica e si è guadagnata molti consensi a livello mondiale nel suo settore. Grazie alla qualità dei suoi prodotti ed alla rinnovata struttura organizzativa, **MBDA** è diventata leader a livello mondiale in un periodo di tempo di soli 5 anni. Dal 2001 **MBDA** ha raddoppiato il suo giro d'affari e consolidato il proprio portafoglio ordini. La società ha inoltre lanciato alcuni programmi basati sulle più avanzate ed innovative tecnologie. Il successo di **MBDA** può essere attribuito, non soltanto alla capacità della società di far lavorare insieme i migliori ingegneri europei nel proprio campo (una capacità supportata e accresciuta anche dalla creazione di alcuni centri d'eccellenza), ma anche grazie allo sviluppo di un vero spirito di collaborazione tra i più di **10.000 dipendenti** del gruppo che lavorano nei vari siti.

Introduzione

Gli argomenti che seguono sono intesi come proposte e linee guida. L'assegnazione dell'argomento della tesi è di competenza del dipartimento stesso.

I temi di ricerca sono incentrati su problematiche legate sia ai sistemi operativi, con particolare attenzione al mondo open source, sia alla produzione del SW in ambito industriale affrontando quindi tutte le problematiche di performance, security, sviluppo e test tipiche di questo ambito.

Le attività di ricerca proposte di seguito sono proposte dal gruppo **IRAD & Innovation** della funzione **Software Engineering and Technologies** di **MBDA**. **IRAD & Innovation** è parte integrante della comunità Software **MindSh@re** di *Finmeccanica*.

Diagnostic Framework

Un framework è una piattaforma reale e concettuale dove codice comune con funzionalità generiche può essere selettivamente specializzato per la realizzazione di un determinato software. La proposta di tirocinio consiste nello sviluppo di un'architettura modulare che consenta riuso dei prodotti per diagnostica sviluppati all'interno di **MBDA**. Lo stage propone uno studio delle tecnologie utilizzate nel contesto dei software per diagnostica in **MBDA** con il fine di ottimizzare l'industrializzazione della linea di prodotto. Peculiarità del lavoro di tesi sarà l'approfondimento delle seguenti tecnologie: protocollo SNMP, Framework Qt, programmazione object oriented, Linux OS.

Motore grafico per GIS 3D

La tesi si dovrà occupare di ricercare nel panorama tecnologico dei software open source e commerciali, ma con supporto agli open format, una soluzione che permetta la visualizzazione di mappe geografiche e relativi layer informativi in 2D e che permetta agevolmente di passare ad una rappresentazione 3D. Il motore grafico dovrà permettere la gestione coerente in termini di geolocalizzazione di oggetti custom da visualizzare e si dovrà definire una ergonomia di visualizzazione 3D. Un fattore chiave da considerare è rappresentato dalle performance dell'intero motore grafico e dalla architettura dei dati, che devono risiedere in un ambiente chiuso, non accessibile a reti esterne. E' previsto lo sviluppo di un dimostratore tecnologico. Le tecnologie di riferimento prevedono l'uso di Linux come OS, Java, C++ come linguaggi di sviluppo.

System Software Security su FINX-SE

In ambito Linux, sono state proposte diverse soluzioni, quali SELinux e grsecurity, che ne hanno elevato il livello di sicurezza e che sono state adottate dalle più importanti distribuzioni commerciali. In particolare, SELinux è stato sviluppato dal DoD americano e propone il meccanismo MAC (Mandatory Access Control) che realizza un controllo di accesso mediante l'assegnazione degli attributi di *dominio* e *ruolo* ad un soggetto (utente o processo) mentre, grsecurity, sviluppato dalla comunità Open Source, limita lo spazio di indirizzamento ed utilizza liste di accesso (ACL: Access Control Lists) per e l'assegnazione di risorse ad un soggetto. Lo studio proposto prevede l'approfondimento di diverse soluzioni che consentano di aumentare la Sicurezza del sistema operativo FINX-SE sviluppato all'interno del Gruppo Finmeccanica. Sintetizzando le proposte di tesi si focalizzano nei punti che seguono:

1. studio dei moduli per la sicurezza offerti dalle patch kernel SELinux e grsecurity: funzionalità, analisi architetturale, implementazione e performance.
2. implementazione del meccanismo di sicurezza MAC (Mandatory Access Control) per il controllo di accesso basato su ruolo e dominio.
3. implementazione del meccanismo di sicurezza ACL (Access Control List) e PaX per il controllo di accesso basato su liste e per la protezione dello spazio di indirizzamento visibile da un processo
4. tecniche di penetration testing, security audits, intrusion prevention ed intrusion detection

Industrializzazione distribuzione Linux per Finmeccanica

Il progetto Linux Tools si propone di realizzare un ambiente di sviluppo completo per la programmazione in C e C++ su sistema Linux. Il fine del progetto è la realizzazione di un IDE comune con funzionalità di debug che riesca ad integrare strumenti di sviluppo comuni e "popolari" come la GNU Autotools, Valgrind, OProfile, RPM, Systemtap, ftrace, etc con progetti allo stato dell'arte. Il lavoro di tesi si propone di analizzare lo stato del progetto (maturità, documentazione, etc) e l'implementazione di un processo di sviluppo legato all'utilizzo del Linux Tool Project finalizzato alla realizzazione di un caso di utilizzo pratico che consenta di analizzare le diverse fase di sviluppo del software: Progettazione, Sviluppo, Integrazione, Test.

Software Reverse Engineering: Generazione automatica di modelli di progettazione per applicazioni software a partire dal codice

Studio dello stato dell'arte di tool di sviluppo software industriali (Artisan, Rhapsody, PRQA) per l'analisi del codice sorgente e la generazione (reverse) di un modello a partire dal codice. Valutazione delle possibilità di estensione dei tool studiati al fine di automatizzare l'analisi del codice di un kernel Linux in ottica certificazione safety/security. L'obiettivo iniziale si concentra sulla realizzazione di uno strumento per generare un modello a partire da porzioni di codice opportunamente definite, per poi individuare delle regole che consentano la modellazione di codice Open Source.

Prestazioni dei Data Distribution Service in sistemi mission critical

Data Distribution Service (DDS) per sistemi real-time è uno standard OMG che ha lo scopo di realizzare una infrastruttura software di comunicazione basata sul paradigma Publish/Subscribe (P/S) con l'obiettivo di garantire caratteristiche quali: scalabilità, real-time, elevate prestazioni, interoperabilità, consistenza dei dati scambiati fra publisher e subscriber. Lo scopo dello studio proposto è un'analisi dello stato dell'arte di questa tecnologia con sperimentazione su sistemi per il controllo del traffico aereo (ATC) basata sull'uso di prodotti COTS e Open Source che implementano il DDS, approfondendo in particolar modo le prestazioni relative alle connessioni RF e WAN.

Parallel programming

La forte evoluzione dell'HW è verso sistemi multi-core (mega-core) dove le CPU e le GPU aumentano il numero dei core riducendone la frequenza. Il tema ha lo scopo di studiare gli impatti della programmazione parallela per trovare il miglior compromesso tra performance sull'HW e riuso di codice legacy. Il scopo è un primo studio sullo stato dell'arte sulla programmazione parallela implicita ed esplicita, funzionale e non funzionale. Successivamente si passerà ad una fase di analisi comparando diverse soluzioni (parallel design patterns, skeleton, openCL, FastFlow, etc) in un conteso real time, mission critical e con vincoli di riuso di codice legacy. Successivamente si dovranno sperimentare le tematiche di programmazione parallela esplicita e quelle implicite identificando le golden rule da seguire nei due casi.

Co-design di algoritmi di ricerca operativa

Nell'ambito del design e dello sviluppo di algoritmi complessi (es. ricerca operativa) si vuole sperimentare un nuovo linguaggio di specifica e sviluppo noto come Active Design Language (ADL). Lo scopo della tesi è quello di studiare il linguaggio e identificare la miglior soluzione di co-design (sviluppo di due team paralleli). Allo stesso modo dovranno essere prese in conto i pattern di programmazione parallela esplicita inclusi gli skeleton che rispondono alla domanda "Cosa devo fare" (lato system programmer) e le tecniche di codifica su hw eterogenei "Come lo devo fare" (lato application programmer) che tenga in conto aspetti non-funzionali (performance, uso memoria, fault tolerance, sicurezza, etc). Gli output della tesi (delle tesi) sono:

- risultati della sperimentazione tramite tool Rhapsody per verificare la bontà del codice così autogenerato
- processo di sviluppo di co-design (Ingegneria del SW)

Virtualizzazione per sistemi real time

Nell'ambito dei sistemi real time nasce la necessità di soddisfare in un unico prodotto requisiti di partizionamento dei sistemi per permettere l'esecuzione contemporanea di più istanze di un generico applicativo sullo stesso nodo fisico. I motivi nascono da ambiti apparentemente disgiunti:

- **Mission critical.** I sistemi mission critical devono introdurre concetti di ridondanza (caldo/freddo) per portare a termine la missione a cui sono preposti
- **Safety critical.** Il partizionamento spaziale e temporale (ARINC653) eleva la robustezza del sistema nel suo complesso in quanto quando una partizione si danneggia (Es. un page fault) le altre continuano a funzionare correttamente senza esserne danneggiate
- **Security.** I sistemi Multiple Independent Level Security (MILS) si basano su concetti di separazione spaziale e sono particolarmente utili come elementi di disaccoppiamento di differenti livelli di sicurezza in sistemi interconnessi.

Lo scopo della tesi è quello di analizzare tutte le tecniche di virtualizzazione basate su Linux includendo para virtualization, full virtualization, container virtualization etc.

Successivamente si dovranno sperimentare le soluzioni su una piattaforma Linux verificando la bontà della soluzione in termini di performance, isolamento, complessità, manutenibilità, etc.

Security information and event management per sistemi mission critical (SIEM)

Nell'ambito di sistemi interconnessi mission critical la problematica della sicurezza è molto importante. Per quanto i vari nodi possano essere certificati secondo ben noti standard di sicurezza, la loro interconnessione non fornisce garanzia di protezione, introducendo la possibilità che una minaccia possa propagarsi. I sistemi SIEM si occupano di collezionare i log di sicurezza dei nodi e visualizzarli. Lo scopo della tesi è quello di fare uno stato dell'arte sui sistemi SIEM e valutarne l'impatto su private network. La tesi dovrà poi sperimentare le diverse soluzioni OpenSource per identificare quella che più si adatta a rete eterogenee private e quindi sviluppare una interfaccia customizzata per la miglior visualizzazione della propagazione della minaccia.

Software Defined Network (SDN) in sistemi mission critical

Le SDN sono sistemi dove il concetto di gestione della rete si separa da quello di gestione del dato. Nei sistemi tradizionali i router o switch concentrano i due piani. Le SDN li separano allocando la gestione della rete in uno o più nodi separati così da permettere una riconfigurazione in tempo reale della rete. Questa capacità è particolarmente importante nel caso di attacchi di sicurezza dove alla propagazione di una minaccia si vuole reagire tramite l'isolamento di una certa sotto rete e far sì che il traffico sensibile venga deviato verso sistemi rilevati sicuri. Lo scopo della tesi è analizzare le soluzioni open source che implementano le SDN (es. OpenFlow) per reti eterogenee (wired, wireless ground, satellitari, data link). Dopo la fase di analisi, si passerà alla fase sperimentale per misurare la bontà delle soluzioni individuate in scenari rappresentativi.

Linux High Availability

I sistemi mission critical e safety critical hanno necessità di servizi di alta affidabilità. Lo scopo della tesi è analizzare le capacità fornite dal progetto Linux HA e verificarle in contesti dove è richiesta la ridondanza delle applicazioni. La tesi dovrà produrre sia una analisi di dettaglio sia una

dimostrazione delle capacità in vari scenari operativi dove verranno simulati dei malfunzionamenti (fault injection).

Contatti

MBDA Italia

Software Engineering Technologies

IRAD & Innovation

Roma Via Tiburtina Km 12,400

Luca Recchia

luca.recchia@mbda-systems.com