E6895 Advanced Big Data Analytics Lecture 12:

GPU and CUDA C

Ching-Yung Lin, Ph.D.
Adjunct Professor, Dept. of Electrical Engineering and Computer Science
IBM Chief Scientist, Graph Computing Research
CUDA: Compute Unified Device Architecture
2001: NVIDIA’s GeForce 3 series made probably the most breakthrough in GPU technology
— the computing industry’s first chip to implement Microsoft’s then-new Direct 8.0 standard;
— which required that the compliant hardware contain both programmable vertex and programmable pixel shading stages

Early 2000s: The release of GPUs that possessed programmable pipelines attracted many researchers to the possibility of using graphics hardware for more than simply OpenGL or DirectX-based rendering.

— The GPUs of the early 2000s were designed to produce a color for every pixel on the screen using programmable arithmetic units known as pixel shaders.

— The additional information could be input colors, texture coordinates, or other attributes
2006:  GPU computing starts going for prime time
— Release of CUDA
— The CUDA Architecture included a unified shader pipeline, allowing each and every arithmetic logic unit (ALU) on the chip to be marshaled by a program intending to perform general-purpose computations.

Example of CUDA processing flow
1. Copy data from main mem to GPU mem
2. CPU instructs the process to GPU
3. GPU execute parallel in each core
4. Copy the result from GPU mem to main mem
Examples

Medical Imaging
Computational Fluid Dynamics
Environmental Science
GPU on a MacBook

NVIDIA GeForce GT 750M:
— 2 * 192 CUDA cores
— max thread number: 2 * 2048
Announcing New Amazon EC2 GPU Instance Type

Posted On: Nov 4, 2013

We are excited to announce G2 instances, a new Amazon Elastic Compute Cloud (EC2) instance type designed for applications that require 3D graphics capabilities. The new instance is backed by a high-performance NVIDIA GPU, making it ideally suited for video creation services, 3D visualizations, streaming graphics-intensive applications, and other server-side workloads requiring massive parallel processing power. With this new instance type, customers can build high-performance DirectX, OpenGL, CUDA, and OpenCL applications and services without making expensive up-front capital investments.

Customers can launch G2 instances using the AWS console, Amazon EC2 command line interface, AWS SDKs and third party libraries. Customers can launch the new instances in the US East (N. Virginia), US West (N. California), US West (Oregon), and EU (Ireland). In addition to On-Demand Instances, customers can also purchase instances as Reserved and Spot Instances. To learn more about G2 instances, visit http://aws.amazon.com/ec2. To get started immediately, visit the AWS Marketplace for GPU machine images from NVIDIA and other Marketplace sellers.
CUDA supports most Windows, Linux, and Mac OS compilers

For Linux:
  Red Hat
  OpenSUSE
  Ubuntu
  Fedora
Hello World!!

```c
#include "../common/book.h"

int main( void ) {
    printf( "Hello, World!\n" );
    return 0;
}
```

Host: CPU and its memory  
Device: GPU and its memory
A Kernel Call

```c
#include <iostream>

__global__ void kernel( void ) {
}

int main( void ) {
    kernel<<<1,1>>>();
    printf( "Hello, World!\n" );
    return 0;
}
```

nvcc handles compiling the function kernel()
it feeds main() to the host compiler
#include <iostream>
#include "book.h"

__global__ void add( int a, int b, int *c ) {
    *c = a + b;
}

int main( void ) {
    int c;
    int *dev_c;
    HANDLE_ERROR( cudaMalloc( (void**)&dev_c, sizeof(int) ) );

    add<<<1,1>>>( 2, 7, dev_c );

    HANDLE_ERROR( cudaMemcpy( &c,
                                dev_c,
                                sizeof(int),
                                cudaMemcpyDeviceToHost ) );

    printf( "2 + 7 = %d\n", c );
    cudaFree( dev_c );

    return 0;
}
Parallel Programming in CUDA C

**Figure 4.1** Summing two vectors

**CPU Vector Sums**
#include "../common/book.h"

#define N 10

void add( int *a, int *b, int *c ) {

    int tid = 0; // this is CPU zero, so we start at zero
    while (tid < N) {
        c[tid] = a[tid] + b[tid];
        tid += 1; // we have one CPU, so we increment by one
    }
}

int main( void ) {

    int a[N], b[N], c[N];

    // fill the arrays 'a' and 'b' on the CPU
    for (int i=0; i<N; i++) {
        a[i] = -i;
        b[i] = i * i;
    }

    add( a, b, c );

    // display the results
    for (int i=0; i<N; i++) {
        printf( "%d + %d = %d\n", a[i], b[i], c[i] );
    }

    return 0;
}
Executing on each of the two CPU cores

CPU CORE 1

```c
void add( int *a, int *b, int *c )
{
    int tid = 0;
    while (tid < N) {
        c[tid] = a[tid] + b[tid];
        tid += 2;
    }
}
```

CPU CORE 2

```c
void add( int *a, int *b, int *c )
{
    int tid = 1;
    while (tid < N) {
        c[tid] = a[tid] + b[tid];
        tid += 2;
    }
}
```
#include "../common/book.h"

#define N 10

int main( void ) {
    int a[N], b[N], c[N];
    int *dev_a, *dev_b, *dev_c;

    // allocate the memory on the GPU
    HANDLE_ERROR( cudaMalloc( (void**)&dev_a, N * sizeof(int) ) );
    HANDLE_ERROR( cudaMalloc( (void**)&dev_b, N * sizeof(int) ) );
    HANDLE_ERROR( cudaMalloc( (void**)&dev_c, N * sizeof(int) ) );

    // fill the arrays 'a' and 'b' on the CPU
    for (int i=0; i<N; i++) {
        a[i] = -i;
        b[i] = i * i;
    }

    // copy the arrays 'a' and 'b' to the GPU
    HANDLE_ERROR( cudaMemcpy( dev_a, a, N * sizeof(int),
                                       cudaMemcpyHostToDevice ) );
    HANDLE_ERROR( cudaMemcpy( dev_b, b, N * sizeof(int),
                                       cudaMemcpyHostToDevice ) );
add<<<N,1>>>( dev_a, dev_b, dev_c );

// copy the array 'c' back from the GPU to the CPU
HANDLE_ERROR( cudaMemcpy( c, dev_c, N * sizeof(int),
                                       cudaMemcpyDeviceToHost ) );

// display the results
for (int i=0; i<N; i++) {
    printf( "%d + %d = %d\n", a[i], b[i], c[i] );
}

// free the memory allocated on the GPU
cudaFree( dev_a );
cudaFree( dev_b );
cudaFree( dev_c );

return 0;
__global__ void add( int *a, int *b, int *c ) {
    int tid = blockIdx.x;       // handle the data at this index
    if (tid < N)
        c[tid] = a[tid] + b[tid];
}
GPU Blocks

**BLOCK 1**

```c
__global__ void
add( int *a, int *b, int *c ) {
    int tid = 0;
    if (tid < N)
        c[tid] = a[tid] + b[tid];
}
```

**BLOCK 2**

```c
__global__ void
add( int *a, int *b, int *c ) {
    int tid = 1;
    if (tid < N)
        c[tid] = a[tid] + b[tid];
}
```

**BLOCK 3**

```c
__global__ void
add( int *a, int *b, int *c ) {
    int tid = 2;
    if (tid < N)
        c[tid] = a[tid] + b[tid];
}
```

**BLOCK 4**

```c
__global__ void
add( int *a, int *b, int *c ) {
    int tid = 3;
    if (tid < N)
        c[tid] = a[tid] + b[tid];
}
```
#include "../common/book.h"

#define N 10

__global__ void add( int *a, int *b, int *c ) {
    int tid = threadIdx.x;
    if (tid < N)
        c[tid] = a[tid] + b[tid];
}

int main( void ) {
    int a[N], b[N], c[N];
    int *dev_a, *dev_b, *dev_c;

    // allocate the memory on the GPU
    HANDLE_ERROR( cudaMalloc( (void**)&dev_a, N * sizeof(int) ) );
    HANDLE_ERROR( cudaMalloc( (void**)&dev_b, N * sizeof(int) ) );
    HANDLE_ERROR( cudaMalloc( (void**)&dev_c, N * sizeof(int) ) );

    // fill the arrays 'a' and 'b' on the CPU
    for (int i=0; i<N; i++) {
        a[i] = i;
        b[i] = i * i;
    }
}
// copy the arrays 'a' and 'b' to the GPU
HANDLE_ERROR( cudaMemcpy( dev_a,
    a,
    N * sizeof(int),
    cudaMemcpyHostToDevice ) );

HANDLE_ERROR( cudaMemcpy( dev_b,
    b,
    N * sizeof(int),
    cudaMemcpyHostToDevice ) );

add<<<1,N>>>( dev_a, dev_b, dev_c );

// copy the array 'c' back from the GPU to the CPU
HANDLE_ERROR( cudaMemcpy( c,
    dev_c,
    N * sizeof(int),
    cudaMemcpyDeviceToHost ) );

// display the results
for ( int i=0; i<N; i++ ) {
    printf( "%d + %d = %d\n", a[i], b[i], c[i] );
}

// free the memory allocated on the GPU
cudaFree( dev_a );
cudaFree( dev_b );
cudaFree( dev_c );

return 0;
### Blocks and Threads

<table>
<thead>
<tr>
<th></th>
<th>Thread 0</th>
<th>Thread 1</th>
<th>Thread 2</th>
<th>Thread 3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Block 0</td>
<td>Thread 0</td>
<td>Thread 1</td>
<td>Thread 2</td>
<td>Thread 3</td>
</tr>
<tr>
<td>Block 1</td>
<td>Thread 0</td>
<td>Thread 1</td>
<td>Thread 2</td>
<td>Thread 3</td>
</tr>
<tr>
<td>Block 2</td>
<td>Thread 0</td>
<td>Thread 1</td>
<td><strong>Thread 2</strong></td>
<td>Thread 3</td>
</tr>
<tr>
<td>Block 3</td>
<td>Thread 0</td>
<td>Thread 1</td>
<td>Thread 2</td>
<td>Thread 3</td>
</tr>
</tbody>
</table>

```c
int tid = threadIdx.x + blockIdx.x * blockDim.x;
```
Figure 5.2 A 2D hierarchy of blocks and threads that could be used to process a 48 x 32 pixel image using one thread per pixel.
Behavior Understanding and Cognitive Security Application
Emerging 'Cognitive Security'

Contextual Situation

Cyber Activities

Structured

Unstructured, Semantic

Cognitive
Goal: Novel inferencing algorithms and system based on multimodality unstructured and structured big data of people

Unstructured and Structured Big Data

~ 200,000 records per person from July 2012 to Jan 2013
Multi-Modality Multi-Layer Understanding of Human Beings

Five Layers of Understanding.
Deterministic Classification + Learning Inference

Sensor Layer
- Available data
  - HR records, Travel records, Badge/Location records, Phone records, Mobile records
  - Transmitted images, speech content, video content

Feature Layer
- Observations
- Hidden states

Concept Layer

Semantics Layer

Cognition Layer
1. **Stealing Login Credentials**: An employee steals usernames and passwords from co-workers and emails them to an outside party.

2. **Exfiltration Prior to Termination**: An employee is leaving the company and decides to take all of their emails and files with them.

3. **Masquerading**: One user is masquerading as another on an unattended workstation.

4. **Bona Fides**: Espionage volunteer prints a bona fides package and takes it to a foreign embassy.

5. **Hiding Undue Affluence**: An employee possesses undue affluence because of ongoing espionage activity. They need to hide the existence of the money from investigators and they perform research on how to do so.

6. **Exfiltration of Sensitive Data Using Screenshots**: An employee steals proprietary/sensitive docs by taking screenshots of specific pages, recursively encrypting the files, and emailing them to a webmail address.

7. **Exfil with Complex Steganography**: An employee uses steganography to hide data in an image file, then uploads that file to a website.

8. **Anomalous Encryption**: A Subject wishes to pass sensitive company information to a foreign government in exchange for that government setting him up with his own business in the foreign country. Subject researches NSA monitoring capabilities with regard to encryption. Subject generates a long random passphrase and stores it in a text document, then tests encrypting and mailing data to their personal account. The subject then exfiltrates sensitive documents by encrypting them with the key and emails the key to an accomplice/handler by including it as an email signature.

9. **Insider Startup**: Three co-conspirators collude to steal company IP. They coordinate the synchronized theft of proprietary information before leaving the company.

10. **Circumventing Sureview**: A user circumvents SureView monitoring to commit a crime.

11. **Masquerading 2**: Subject sets up a rogue SSH server on another user's machine. They also make a copy of the local Windows password file and copy the file off over the network.
12. **Layoff Logic Bomb:** An engineer is worried about rumors of impending layoffs. He feels that he needs some kind of an “insurance policy”, in case he gets laid off or fired. He creates a "logic bomb" which will delete all files from a number of company Linux systems in five days, unless he resets the timer before then. We may or may not observe the subject changing the timer, depending on the scenario variation.

13. **Outsourcer's Apprentice:** (Lifted from [http://www.bbc.co.uk/news/technology-21043693](http://www.bbc.co.uk/news/technology-21043693).) A software developer outsources his job to China and spends his workdays surfing the web. Some surfing activity occurs on his main workstation while the subcontractor is active, but most of it occurs on a second laptop he uses to try to minimize his interference with the subcontractor. He pays just a small fraction of his salary to a company based in China to do his job. The developer provides remote access to his machine by providing his VPN credentials to the Chinese company and enabling Terminal Services on his workstation. The Chinese consulting firm sends the developer PayPal invoices for the work performed, and the developer pays them.

14. **Survivor's Burden:** The subject is disgruntled after his team experienced layoffs (and a logic bomb), greatly increasing his workload. He hopes to become the team lead, but is passed over for the position and takes matters into his own hands by stealing company IP using DropBox.

15. **Manning Up:** Subject has conducted extensive research concerning Bradley Manning, Bradley Manning connected to WikiLeaks, and Bradley Manning's treatment by the U.S. Government. Subject has engaged coworkers concerning Bradley Manning. Subject has been researching the DNS protocol exploits, and specifically how to handcraft DNS queries, including through batch files. Subject has been experimenting with this knowledge, especially handcrafted DNS queries for an unknown purpose.

16. **Manning Up Redux:** Subject has continued with research concerning Bradley Manning, Bradley Manning connected to WikiLeaks, and Bradley Manning's treatment by the U.S. Government. Subject has engaged a co-worker concerning Bradley Manning. Subject has been researching whistleblower laws, sites that accept and post whistleblower material, and confidential reporting mechanisms. Subject has created text files and batch files that export segments of the text files through specially crafted DNS queries.
● **Personal stress:**
  - Gender identity confusion
  - Family change (termination of a stable relationship)

● **Job stress:**
  - Dissatisfaction with work
    - Job roles and location (sent to Iraq)
    - Long work hours (14/7)

● **Unstable Mental Status:**
  - Fight with colleagues, write complaining emails to colleagues
  - Emotional collapse in workspace (crying, violence against objects)
  - Large number of unhappy Facebook posts (work-related and emotional)

● **Planning:**
  - Online chat with a hacker confiding his first attempt of leaking the information

● **Attack:**
  - Brought music CD to work and downloaded/copied documents onto it with his own account
IBM ADAMS Functional Schematic

**Infrastructure**
- Provenance
- Flexible component mashup
- Transparent data source layer

**Data Transformation**

**Data processing**

**Data Storage**

---

**Emotion Analysis**
- Risk Emotion Detection
- Graphical Dynamic Behavior Model
- Scalable Time-Adaptive Local Outlier Detector
- HMM Common Event Identification

**Behavior Analysis**
- Web domain, Query terms, topic analysis
- Social communication graph analysis
- Scalable graph feature computation
- Action causality group anomaly

**Semantic Analysis**

**Social Analysis**

**Fusion**

**Reasoning / Risk Prediction**
- Large-Scale Latent Markovian Bayesian Network

**Multimodality Anomaly Detection & Exploration User Interface**

**Visual Analysis**
Computational deep learning is closely related to theories of brain development.

A generative graphical model with many layers of random variables:
- Relationship among features are captured by higher level latent variables.
- Trained in a layer-wise manner, fine with deeper architectures.
- Suitable for multimodal inference.
  - Allows many latent layers without increasing training complexity.
  - Allows multiple level abstraction/representation of observable features.

Ultimate causation (detection output): Multi-layer abstraction of observable features.
Concept Layer Example – Finding Outliers

- Directly Derivable from Features
- Surfacing unusual search terms, webpage domains, email communications, type of browsed webpages, etc.
- Comparing with:
  - Self
  - Peer Groups
  - All populations

Local-density in Multi-dimensional Feature Space

Comparing with Self or others: change of local-density
Feature Layer and Concept Layer – ~70 detectors

- **Local outlier factor with time (LOF)**
  - Identify users whose neighborhood density is different from his/her neighbor's neighborhood density, where the neighbors are defined as
    - The user's past history
    - The whole population's past history
    - Compare with Peer groups: (1) people who this user send most emails, (2) IM, (3) share the same machine (4) access the same file
  - Data type applied: number of file copied per day, number of pages printed per day, log on time each day, length of each session on each machine each day

- **Graph feature outlier**
  - Identify users whose egonet features are different from the user's past history with Gaussian mixture model
  - Egonet features: all/external/internal email exchange, number of senders/recipient, number of email receive/sent, new senders/recipient, personal/spam emails, use BCC

- **Graph change**: identify the user whose email communication pattern changes a lot from the previous week

- **Rareness**
  - Identify users who have rare actions compared with the user's own history or the whole community
  - Action types: search keywords on websites, access/upload to/download from rare http websites, send emails to rare external email domain, access machines that the user rarely used before, use process that is rare regarding to the user and the whole population

- **High file path number**: access files from abnormally large number of different source devices

- **Overlapping PC**: access another user's primary machine within a short time frame after the primary user logged out
Classify each e-mail thread into one or more of 12 categories (anger, aggressiveness, anxiety, disgruntlement, possessiveness/territorial, sexuality, entitlement, negative emotionality, dehumanization, ideological expression, victimization, and depression), overall risk level and personal vs. business email

Example:

I had an awful day today, John stormed into my office today!!!!. I am exceedingly resenting this behavior by my manager. He IS such an AIRHEAD. What shall I do? Alice

*I had an awful day today, *John stormed into my office today!!!!. *I am exceedingly resenting this behavior by my manager. *He IS such an AIRHEAD. *What shall I do? *Alice

I had an awful day today, John stormed into my office today!!!!. I am exceedingly resenting this behavior by my manager. He IS such an AIRHEAD. What shall I do? Alice

# of rules hit = 1
# of lexical features: = 3
# of specific concepts = 1

Weighted and normalized score on Disgruntlement = 0.34
• **Time adaptive fusion**: fuse the user’s termination time, output of final Pareto depth analysis, and user's rare http access

• **Pareto depth analysis**
  - Fuse multi-modal detection inputs to determine the cause
  - Domains applied: all LOF output, hierarchically fuse low-level detectors according to data type and fuse all data type

• **Email emotion detection**
  - Rule-based engine to detect emotion from emails the users send each day
  - Emotion category: anger, aggression, anxiety, possessiveness, ideological expression, depression, sexuality,

• **Email emotion fusion**: fuse the emotion categories into the risk level of the user this day

• **File access sequence detection**: identify anomalous sequence of actions applied on files. For example: download the file, edit it, and send out to rare external email recipients

• **Markovian action transition detection**:
  - Identify unusual transition time between actions and unusual action transition
  - Example: user download a software from the website and after 1.5 second the software is used on an image
Concept Layer Detectors

- **Hidden Markov Model**
  - Combine all low level detectors and train a hidden markov model to detect abnormal sequence of rare actions
  - Filter out community events

- **Frequency Domain Analysis**
  - Transform the action time series into frequency domain and detect outliers with Gaussian mixture model
  - Action types: website visit, device access, email exchange, logon machines

- **Conditional Random Field**
  - Use all data types to identify rare sequence of actions
Framework for Scalable Graphical Models

Big Data Problems

Bayesian Network
Evidence

Inference
Junction Tree
Evidence Propagation
Parallel Computation Kernels
Posterior probability of query variables

Output

Parallel Inference in Graphical Model
From graph representation to parallel execution

Middleware and Runtime Optimization
Generic Graph Interface with Runtime library support
IBM PAMI & RDMA
Architecture-aware Schedulers

Parallel Computing Platforms
UltraSPARC T3 Processor Diagram
Nehalem
Multicore
Manycore
Cluster

Parallelization at multiple granularity levels
Our Prior Work on Large-Scale Bayesian Networks

**Approach:** BN is built to represent the dependence among sensors and replicated across timesteps. BN parameters are learned from over 15 years of ARM climate data to support distributed climate sensor validation. Inference validates sensors in the connected instruments.

---

**Bayesian Network**
- 3 timesteps
- 63 variables
- 3.9 avg states
- 4.0 avg indegree
- 16,858 CPT entries

**Junction Tree**
- 67 cliques
- 873,064 PT entries in cliques
Graphical Computation

- **Step 1:** Computations on graph structures / topologies
  - Example → converting Bayesian network into junction tree, graph traversal (BFS/DFS), etc.
  - Characteristics → Poor locality, irregular memory access, limited numeric operations

Bayesian network to Junction tree

- **Step 2:** Computations on graphs with rich properties
  - Example → Belief propagation: diffuse information through a graph using statistical models
  - Characteristics
    - Locality and memory access pattern depend on vertex models
    - Typically a lot of numeric operations
    - Hybrid workload

- **Step 3:** Computations on dynamic graphs
  - Characteristics
    - Poor locality, irregular memory access
    - Operations to update a model (e.g., cluster, sub-graph)
    - Hybrid workload

Score core subgraph
Parallel Inference – From Bayesian Network to Junction Tree

- Conditional dependence among random variables allows information propagated from a node to another \(\Rightarrow\) foundation of probabilistic inference

Given evidence (observations) \(E\), output the posterior probabilities of query \(P(Q|E)\)

Bayes’ theorem can not be applied directly to non-singly connected networks, as it would yield erroneous results

Therefore, junction trees are used to implement exact inference

\[\Psi(8) = P(8)\]

\[\Psi(3,1,2) = P(3|1,2)P(1)P(2)\]

\[\Psi(3,2,1) = P(3|1,2)P(1)P(2)\]

\[\Psi(3,2) = P(3|2)P(2)\]

\[\Psi(2,1) = P(2|1)P(1)\]

\[\Psi(1) = P(1)\]
Parallel Inference – Parallelism at Multiple Granularity Levels

Each of the following arrays has $r^w$ entries

- \( (a=0, b=0, c=0, d=0, e=1, f=1) \)
- \( (a=0, b=0, c=0, d=1, e=0, f=0) \)

\[ \Psi_C = \begin{bmatrix} 0.21354 & 0.01219 & \cdots \end{bmatrix} \]

Only this array is stored with clique C

Marginalization

- Absorb
- Marginalize
- Extend
- Multiply/Divide

\[ \psi_S^* = \sum_{\forall s \not\in S} \psi_S^* \]
\[ \psi_X = \psi_X \frac{\psi_S^*}{\psi_S} \]

Processor 1

Processor 2

© 2013 IBM Corporation
Parallel Inference – Architecture-aware Parallelization

Sync. overhead

Wait-free data structure

Thread 1
Thread 2
Thread P
12. **Layoff Logic Bomb**: An engineer is worried about rumors of impending layoffs feels that he needs some kind of an “insurance policy”, in case he gets laid-off or fired. He creates a "logic bomb" which will delete all files from a number of company Linux systems in five days, unless he resets the timer before then. We may or may not observe the subject changing the timer, depending on the scenario variation.

13. **Outsourcer’s Apprentice**: (Lifted from http://www.bbc.co.uk/news/technology-21043693.) A software developer outsources his job to China and spends his workdays surfing the web. Some surfing activity occurs on his main workstation while the subcontractor is active, but most of it occurs on a second laptop he uses to try to minimize his interference with the subcontractor. He pays just a small fraction of his salary to a company based in China to do his job. The developer provides remote access to his machine by providing his VPN credentials to the Chinese company and enabling Terminal Services on his workstation. The Chinese consulting firm sends the developer PayPal invoices for the work performed, and the developer pays the.

8. **Anomalous Encryption**: A Subject wishes to pass sensitive company information to a foreign government in exchange for that government setting him up with his own business in the foreign country. Subject researches NSA monitoring capabilities with regard to encryption. Subject generates a long random passphrase and stores it in a text document, then tests encrypting and mailing data to their personal account. The subject then exfiltrates sensitive documents by encrypting them with the key and emails the key to an accomplice/handler by including it as an email signature.
IBM ADAMS System Type 1: Espionage and IP Theft

- Sequence of actions: plan to leave the company (dissatisfaction) -> look for / land another job -> steal information (large amount) -> leave
- Ways of transferring information
  - Email, phone, fax, remote download, download to removable drives, printing
- Reason
  - Job-related dissatisfaction
  - Feel entitled to take the information
    - credentials, source code, client information
    - Should identify from the content
  - Loyalty to home country
  - To benefit next job

The detectables:
- On the transfer path:
  - Email to competitors/webmails/foreign countries (with large attachments), copied to removable drives (CD, USB, removable HD), and then delete same files on the server
  - Total size of file access per day
- On reasoning: suspicious foreign travel/information gathering
- On planning: curiosity towards information outside of his area
IBM ADAMS System Type 2: Sabotage

- Sequence of actions: behavior precursors -> technical precursors -> attack (mostly occur after the insider leave the organization)
  - Reason: disgruntled
    - demotion/no promotion
    - supervisor/coworker disagreement
    - accessed changed (entitlement)
    - bonus/salary dissatisfaction
    - stress under deadlines/milestones
    - layoffs

- Behavior precursors (all cases)
  - Preceded by stressful events
  - job performance decline, information gathering, reduced work hours
- Technical precursors
  - Followed by termination
  - Half of the cases occur within work hour, and more than half of the cases use remote log in
  - Creating backdoors, installing remote network admin tool, disable anti-virus, installing malicious tools (password cracker, virus), delete/download backups
  - Account used: shared account, compromised account, account for external users
- Sequence of actions: significant financial problems -> (in contact with outsiders) -> exfiltrate/edit information in critical files
- The information manipulated are usually small and over a long period of time
- Reasons: financial problems
  - Medical bills
  - Addiction
  - Expensive tastes
  - Family
  - Treat by outsiders
Fusion from IBM ADAMS Single-Modal Detectors

• Comparison bases:
  - Whole community / colleagues
  - Personal history
• Anomaly detection methods
  - Time-adaptive local outlier factors
    • Feed in datapoints sequentially by time => capture behavior transition
  - Bi-metric (number of access and number of user in the community) statistic-based detection
• Semantics
  - Web access
    • type of downloaded data (ex: image, video), categories of web domain (travel, webmail, etc), query terms, SureView-defined access methods (ex: HTML, Javascript)
  - Process
    • Threat programs (virus protection disabled), development environments (Linux simulator), secret transmission (encryption, cryptography)
  - Email domains
  - File access
    • Print files from network, download files to removable drives
  - Risk-related emotion and stressor
Use Case 1.1: Espionage/IP Theft Scenario

Description:
- Bona Fides: Espionage volunteer prints a bona fides package and takes it to a foreign embassy

- Preparation: collect valuable information: download, print

- Attack

- Motivation: Job dissatisfaction. Loyalty to home country

- Entitlement

- Communication Precursor: transfer information externally. Contact competitors

- Escape plan: plan foreign travel, archive/encrypt data

- Web access
  - type of downloaded data (ex: image, video), categories of web domain (travel, webmail, etc), query terms, SureView-defined access methods (ex: HTML, Javascript)

- Process
  - Threat programs (virus protection disabled), development environments (Linux simulator), secret transmission (encryption, cryptography)

- Email domains

- File access
  - Print files from network, download files to removable drives

- Risk-related emotion and stressor
Use Case 1.2: RedTeam Espionage/IP Theft Scenarios

Descriptions:

- Anomalous Encryption: A Subject wishes to pass sensitive company information to a foreign government in exchange for that government setting him up with his own business in the foreign country. Subject researches NSA monitoring capabilities with regard to encryption. Subject generates a long random passphrase and stores it in a text document, then tests encrypting and mailing data to their personal account. The subject then exfiltrates sensitive documents by encrypting them with the key and emails the key to an accomplice/handler by including it as an email signature.

- Web access
  - type of downloaded data (ex: image, video), categories of web domain (travel, webmail, etc), query terms, SureView-defined access methods (ex: HTML, Javascript)

- Process
  - Threat programs (virus protection disabled), development environments (Linux simulator), secret transmission (encryption, cryptography)

- Email domains

- File access
  - Print files from network, download files to removable drives

- Risk-related emotion and stressor

Motivation: Job dissatisfaction. Loyalty to home country
Entitlement
Preparation: collect valuable information: download, print
Communication
Precursor: transfer information externally
Contact competitors
Escape plan: plan foreign travel, archive/encrypt data
termination
Attack
Use Case 2.1: RedTeam Sabotage Scenarios

Descriptions:

- **Layoff Logic Bomb**: An engineer is worried about rumors of impending layoffs and feels that he needs some kind of an “insurance policy”; in case he gets laid-off or fired. He creates a "logic bomb" which will delete all files from a number of company Linux systems in five days, unless he resets the timer before then.

- **Web access**
  - type of downloaded data (ex: image, video), categories of web domain (travel, webmail, etc), query terms, SureView-defined access methods (ex: HTML, Javascript)

- **Process**
  - Threat programs (virus protection disabled), development environments (Linux simulator), secret transmission (encryption, cryptography)

- **Email domains**

- **File access**
  - Print files from network, download files to removable drives

- **Risk-related emotion and stressor**
Use Case 2.2: RedTeam Sabotage Scenarios

Descriptions:
• Manning Up: Subject has conducted extensive research concerning Bradley Manning, Bradley Manning connected to WikiLeaks, and Bradley Manning's treatment by the U.S. Government. Subject has engaged co-workers concerning Bradley Manning. Subject has been researching the DNS protocol exploits, and specifically how to hand craft DNS queries, including through batch files. Subject has been experimenting with this knowledge, especially hand crafted DNS queries for an unknown purpose.

  – Web access
    • type of downloaded data (ex: image, video), categories of web domain (travel, webmail, etc), query terms, SureView-defined access methods (ex: HTML, Javascript)
  – Process
    • Threat programs (virus protection disabled), development environments (Linux simulator), secret transmission (encryption, cryptography)
  – Email domains
  – File access
    • Print files from network, download files to removable drives

Risk-related emotion and stressor
Use Case 3.1: RedTeam Fraud Scenarios

Descriptions:

- Hiding Undue Affluence: An employee possesses undue affluence because of ongoing espionage activity. They need to hide the existence of the money from investigators and they perform research on how to do so.

- Web access
  - type of downloaded data (ex: image, video), categories of web domain (travel, webmail, etc), query terms, SureView-defined access methods (ex: HTML, Javascript)

- Process
  - Threat programs (virus protection disabled), development environments (Linux simulator), secret transmission (encryption, cryptography)

- Email domains
- File access
  - Print files from network, download files to removable drives
- Risk-related emotion and stressor
Use Case 3.2: RedTeam Fraud Scenarios

Descriptions:
• Outsourcer's Apprentice: A software developer outsources his job to China and spends his workdays surfing the web. Some surfing activity occurs on his main workstation while the subcontractor is active, but most of it occurs on a second laptop he uses to try to minimize his interference with the subcontractor. He pays just a small fraction of his salary to a company based in China to do his job. The developer provides remote access to his machine by providing his VPN credentials to the Chinese company and enabling Terminal Services on his workstation. The Chinese consulting firm sends the developer PayPal invoices for the work performed, and the developer pays them.

- Web access
  • type of downloaded data (ex: image, video), categories of web domain (travel, webmail, etc), query terms, SureView-defined access methods (ex: HTML, Javascript)
- Process
  • Threat programs (virus protection disabled), development environments (Linux simulator), secret transmission (encryption, cryptography)
- Email domains
- File access
  • Print files from network, download files to removable drives
- Risk-related emotion and stressor

Preparation:
- manipulate important information. get external connections

Hide traces:
- research on hiding benefits

Communication Precursor:
- share credential information externally

Financial stressors, addiction

Attack
Oct Benchmark Results on the real Data (5,000 people x 0.5M rec. / person)

- Rankings of the red team abnormal insiders out of the ‘5000 people per month based on the ‘Attack’ scores. Each case’s rankings in the 3 IBM ADAMS systems (Sabotage, Espionage, and Fraud) are shown.
- The 'All' ranking is used when the analyst only wants a single ranked list per month. People are ranked based on the highest anomaly scores of a person in these 3 systems.

<table>
<thead>
<tr>
<th></th>
<th>Sabotage</th>
<th>Espionage</th>
<th>Fraud</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dec</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sabotage (Scenario 12)</td>
<td>4</td>
<td>241</td>
<td>1667</td>
<td>9</td>
</tr>
<tr>
<td>Espionage (Scenario 8)</td>
<td>981</td>
<td>1</td>
<td>120</td>
<td>1</td>
</tr>
<tr>
<td>Fraud (Scenario 13)</td>
<td>1526</td>
<td>454</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Jan</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Fraud (Scenario 13)</td>
<td>4230</td>
<td>3367</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>Espionage (Scenario 14)</td>
<td>11</td>
<td>44</td>
<td>574</td>
<td>30</td>
</tr>
<tr>
<td>Feb</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Espionage (Scenario 14)</td>
<td>1936</td>
<td>73</td>
<td>232</td>
<td>203</td>
</tr>
<tr>
<td>Espionage (Scenario 4)</td>
<td>4101</td>
<td>9</td>
<td>803</td>
<td>26</td>
</tr>
<tr>
<td>Sabotage (Scenario 15)</td>
<td>65</td>
<td>4101</td>
<td>654</td>
<td>181</td>
</tr>
<tr>
<td>Mar</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sabotage (Scenario 16)</td>
<td>1</td>
<td>1690</td>
<td>294</td>
<td>1</td>
</tr>
<tr>
<td>Fraud (Scenario 5)</td>
<td>1544</td>
<td>9</td>
<td>5</td>
<td>10</td>
</tr>
<tr>
<td>Espionage (Scenario 4)</td>
<td>4325</td>
<td>11</td>
<td>46</td>
<td>27</td>
</tr>
</tbody>
</table>

12. **Layoff Logic Bomb**: An engineer is worried about rumors of impending layoffs feels that he needs some kind of an “insurance policy”, in case he gets laid-off or fired. He creates a "logic bomb" which will delete all files from a number of company Linux systems in five days, unless he resets the timer before then.

13. **Outsourcer's Apprentice**: A software developer outsources his job to China and spends his workdays surfing the web. Most surfing occurs on a second laptop. He pays just a small fraction of his salary to a Chinese company to do his job. The developer provides his VPN credentials to the company and enabling Terminal Services on his workstation.

8. **Anomalous Encryption**: A Subject wishes to pass sensitive information to a foreign government in exchange of helping set up his own business. Subject researches NSA monitoring capabilities, generates random passphrase, tests encrypting and mails data to personal account. The subject encrypts documents and emails the key.

We significantly outperformed competitors. In the Oct review, our results were: 4 cases as Top 1, 3 cases between Top 3-5, 3 cases between Top 9-44, and 2 cases between Top 50-100. Competitor System 1 could not report results. Competitor System 2 reported: 3 of the 12 cases Top #50-#100, 6 cases Top #101-#500, and 3 cases beyond #500
Results on Vegas Data (~5,000 people): Missing Single-Modality Detectors

- Fusion results with single modality detectors missing (set the detector output to 0)
  - Over all 4 months, based on the rank for target types
  - Highest rank keeps the same with or without the features
  - Without QueryTerm and EmlDomainPersonal, some of the redteam users rank very low (very difficult to detect)
Further Improvement – Analyst Feedback

- To simulate our system incorporating analyst feedback
  - Annotate 5 user-day activity that scores higher than redteam users for each type for 4 months
    - Label the input to fusion as not anomalous/ anomalous
    - Change the input to fusion to 0 if marked as not anomalous
  - Total annotation took one person one day

<table>
<thead>
<tr>
<th>Targeted-Type</th>
<th>Original</th>
<th>Feedback</th>
<th>All</th>
<th>Original</th>
<th>Feedback</th>
</tr>
</thead>
<tbody>
<tr>
<td>Dec Sabotage(12, UC 2.1)</td>
<td>4</td>
<td>1</td>
<td>9</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>Espionage(8, UC 1.2)</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>Fraud(13, UC 3.2)</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Jan Fraud(13, UC 3.2)</td>
<td>1</td>
<td>1</td>
<td>2</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>Espionage(14)</td>
<td>44</td>
<td>43</td>
<td>30</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>Fraud(5, UC 3.1)</td>
<td>3</td>
<td>2</td>
<td>8</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>Feb Espionage(14)</td>
<td>73</td>
<td>73</td>
<td>203</td>
<td>203</td>
<td></td>
</tr>
<tr>
<td>Espionage(4, UC 1.1)</td>
<td>9</td>
<td>9</td>
<td>26</td>
<td>26</td>
<td></td>
</tr>
<tr>
<td>Sabotage(15, UC 2.2)</td>
<td>65</td>
<td>58</td>
<td>181</td>
<td>161</td>
<td></td>
</tr>
</tbody>
</table>