COMS3000/7003

Week 2

Access Control, Authentication, Authorisation

David Ross

I have moved

- Now GP South Room 310, not 304.
  - Consultation by appointment in 78-411 after the lecture each week.

Audio Fixed (I hope)

- Seem to have broken Echo while I was playing with video output on my laptop.
- Last weeks slides contained all the detail so no big issue → but this is not the case for many of the lectures

From the Course Profile

"You are not required to attend any of the teaching sessions (except those in which an assessment activity is taking place), however, you are STRONGLY encouraged to do so. Do not enrol in this course if you will be unable to physically attend most lectures.

...

Failure to attend a session may result in you being disadvantaged."

Assessment

- In-Class Quiz (20%)
  - 20 multiple choice (20%)
  - CLOSED Book
- Assignment (20%)
  - Written Report
  - More details later…
- Final Exam (60%)
  - During examination period at end of semester
  - Open book
- See also Course Profile for details

Assignment

- Next week (Week 3) the assignment will be issued. Details next week.
- Due in Week 9 - 22 Sep 2017 16:00
- STRONGLY recommend you submit before 2 pm while the assignment office is staffed

Outline of Today’s Lecture

- Trust
- Access Control
- Authentication, Authorisation
- Passwords
- Cryptographic one-way hash functions

Important Security Characters – Often used in Examples

- Alice and Bob
  - Originally introduced by Ron Rivest, in 1978
  - They usually want to send messages to each other (the good guys)
- Eve
  - Eavesdropper (passive)

- Mallory, Trudy
  - Malicious Attacker, Intruder (active)
- Trent
  - Trusted 'third party'

- Carol, Dave, …
- (Alice and Bob images by J.F Kurose and K.W. Ross, Computer Networking: A Top-Down Approach)

Who might Alice and Bob be?

- Real-life people
- Web browser & Web server for electronic transactions (e.g., on-line purchases)
- On-line banking client and server
- DNS servers
- Routers exchanging routing table updates
- Other examples?

The Concept of Trust

- The concept of ***Trust*** and ***Trustworthiness*** are fundamental for Information Security.
- **Trust** according to the Oxford Dictionary:
  - *‘confidence, strong belief, in the goodness, strength, reliability of something or somebody’, ‘ responsibility’.*
- What does it mean to trust somebody/something? Can you give an Example?
  - We can trust somebody or something if it/he/she behaves as expected.
  - Bob lends Alice $100 and he trusts her, i.e. he expects her to give it back.

Trust vs. Trustworthiness

- What is the difference between ***trusted*** and ***trustworthy***?
- Can you give an example for:
  - Bob is trusted
  - Bob is trustworthy

- Bob’s boss trusts him to handle large amounts of company money, i.e. Bob is **trusted**.
- If Bob embezzles the money, he is still trusted by his boss, but he is definitively **not trustworthy**.
- It could be the other way round as well.

Direct and Indirect Trust

- Trust can be based on direct experience
  - Example:
    - Bob has conducted business with Alice for several years, therefore he trusts her.
- What if Alice and Bob want to do business but have never met or communicated before? How can they establish a level of trust?
  - Via indirect experience or recommendation.
  - Trent tells Bob that Alice is trustworthy. If Bob trusts Trent, this trust is transferred to Alice. (Trent acts as a so-called *Trusted Third Party*.)
- Can you give a concrete example (e.g. e-Commerce)?
  - *ebay* collects feedback (recommendations) from users and acts as a trusted third party (We assume *ebay* is trustworthy!)
  - This allows creating a level of trust between parties that don't know each other

Information Security

- Information Security is about protecting information assets
- Information assets need to be protected to provide (5 aspects of Information Security):
  - **Confidentiality**
  - **Integrity**
  - **Availability**
  - Authenticity (Integrity of provenance)
  - Non-repudiation (Integrity of action/timeline)

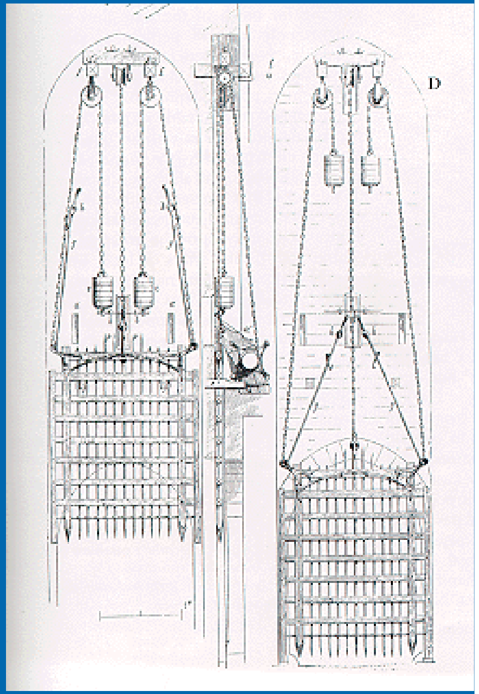
- We need to control who has access to information → **Access Control**
  - As we will see, to achieve this we often need Authentication (or Authenticity)

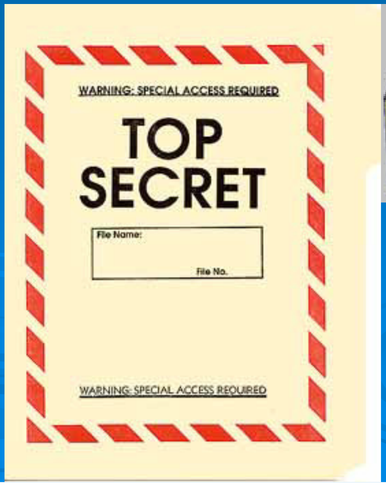
Access Control

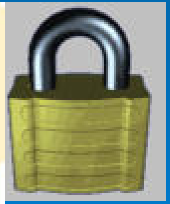
- Lack of Access control can lead to Confidentiality, Integrity and Availability (CIA) being compromised
- We need to control who can read, write and modify data.
- Can you give examples of Access Control in general and in the context of Information Security?

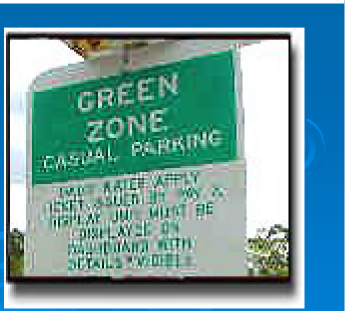
- Examples:
  - Pub (need ID to prove you are over 18)
  - Airport security (Passport and boarding pass required)
  - Withdraw cash from an ATM (PIN)
  - Swipe card access for Computer Labs (Need swipe card)
  - Password to control access to computer account (Need to remember password)
  - Retina scan to access high security military sites
    - (characteristics of human body)
  - Traffic Lights (access depends on time)
  - Cinema (need ticket to get access to movie theatre)

Real Word Access Control




WARNING: SPECIAL ACCESS REQUIRED
TOP SECRET
File Name:
File No.
WARNING: SPECIAL ACCESS REQUIRED





Access Control

- What is an access control decision in Information Security typically based on?
  - Mostly identity
- 2 separate functions:
  - Establish Identity (Identification and Authentication)
    - Identification: Determine identity (person or machine)
      - “Tell me who you are”
    - Authentication: Verify identity (proof)
      - “Prove that you really are who you claim to be”
  - Authorisation
    - Once the identity of a person is known, a decision can be made about granting or denying access
      - Identity is often linked to a ‘Role’ → Role Based Access Control (RBAC)
    - Examples:
      - Once Alice has provided proof of identity via a password, she is granted access to her Web banking account

Access Control

- What about the Access Control example of the Pub or traffic lights?
- What does the Access Decision depend on?
  - Pub:
    - Decision depends on age (attribute, characteristic)
    - -> attribute based access control
  - Traffic Lights:
    - Decision depends on time
  - Cinema
    - Ownership of ticket (anonymous)
- In Information Security, access control is typically based on Identity.
- Identification and Authentication are crucial
- It also allows to create an audit trail → Accountability:
  - "Who did what when?"

Class Exercise
Identification - Authentication

- In teams of 2-4 discuss and write down examples of identity-based Access Control and name the Identification, Authentication and Authorisation mechanisms involved
  - In general
  - In Information Security

- Examples:
  - Border Control at air port
    - Identification → Passport
    - Authentication → verify photo, biometrics
    - Authorisation → e.g. check Visa status
  - Computer Login
    - Identification → Enter your user name
    - Authentication → Enter password
    - Authorisation → check if user has an account, and what role

- What about swipe card access to the computer lab?
  - Identification and Authentication are combined

Authentication

- "The process of verifying a claimed identity."
  - Authentication provides proof of identity

- How can this proof be provided?
  (3 general types of mechanisms)

  - With something you **know**
    - Password, PIN
  - With something you **have**
    - Physical Key, token, smart card …
  - With something you **are (or do)**
    - Unique characteristic of human body
    - e.g. fingerprint, voice, hand geometry … (Biometrics)
    - More on this later

- What's the most common method of authentication?

Passwords

- Most common method for Authentication
  - "something you know"

- Secure?

- Problems with passwords?
  - Users forget passwords
    - Too many passwords to remember
  - Users choose same password for multiple systems
    - If one system is compromised, all systems are compromised
  - Users choose weak or guessable passwords
    - Problem: Easy to remember passwords are often also easy to guess
  - Users write passwords on Post-it notes and stick them to their computer screen
  - Failure to reset default password
  - …

Default Passwords

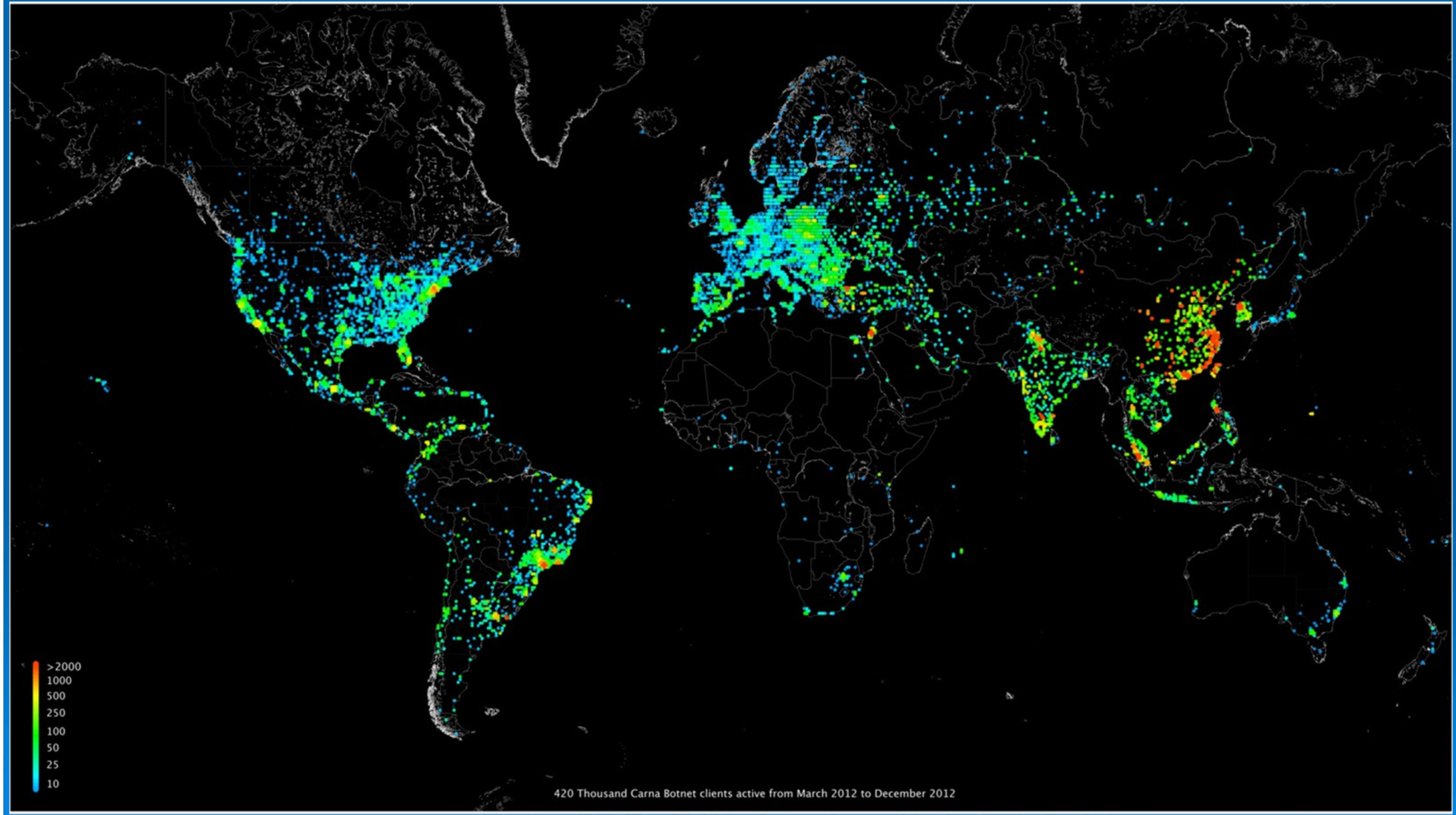
- These days, are there still computers/devices running services with default passwords connected to the Internet? Surely not (?)

- "Carna Botnet"
  - ~ 500,000 devices compromised via services such as Telnet, with default or no passwords
  - Mostly small (embedded) devices, IP cameras, routers, etc. (IoT)
- Purpose
  - "Internet Census 2012"
  - Internet wide port scan of all IPv4 addresses
  - Collection of huge data set

- "*Two years ago while spending some time with the Nmap Scripting Engine (NSE) someone mentioned that we should try the classic telnet login root:root on random IP addresses. This was meant as a joke, but was given a try. We started scanning and quickly realized that there should be several thousand unprotected devices on the Internet.*"
  - http://internetcensus2012.bitbucket.org/paper.html

- BTW, *nmap* is a widely used security scanner or 'port scanner', used for 'Pen Testing'
  - http://nmap.org
  - Penetration Testing
    - ' …attack on a computer system with the intention of finding security weaknesses …'
    - http://en.wikipedia.org/wiki/Penetration_test

Carna Botnet Clients



420 Thousand Carna Botnet clients active from March 2012 to December 2012

… back to Passwords

There are different approaches of dealing with passwords …

"Sorry about the odor. I have all my passwords tattooed between my toes."

Attacks against Passwords

- What are possible attacks to ‘find’ a user’s password?
- Password guessing
  - Brute force, dictionary attacks
  - Online, offline (more on this later)
- Social Engineering
  - You get a phone call from somebody claiming he/she is the sysadmin and needs your password to reset your account
    - “Phishing attacks” → get users’ credentials (identity theft)
- Eavesdropping
  - Some (insecure) applications send password in cleartext over the network (ftp, telnet, http) → “Network sniffer”, e.g. *wireshark*
- “Shoulder surfing”
  - Somebody is watching you when you enter the password

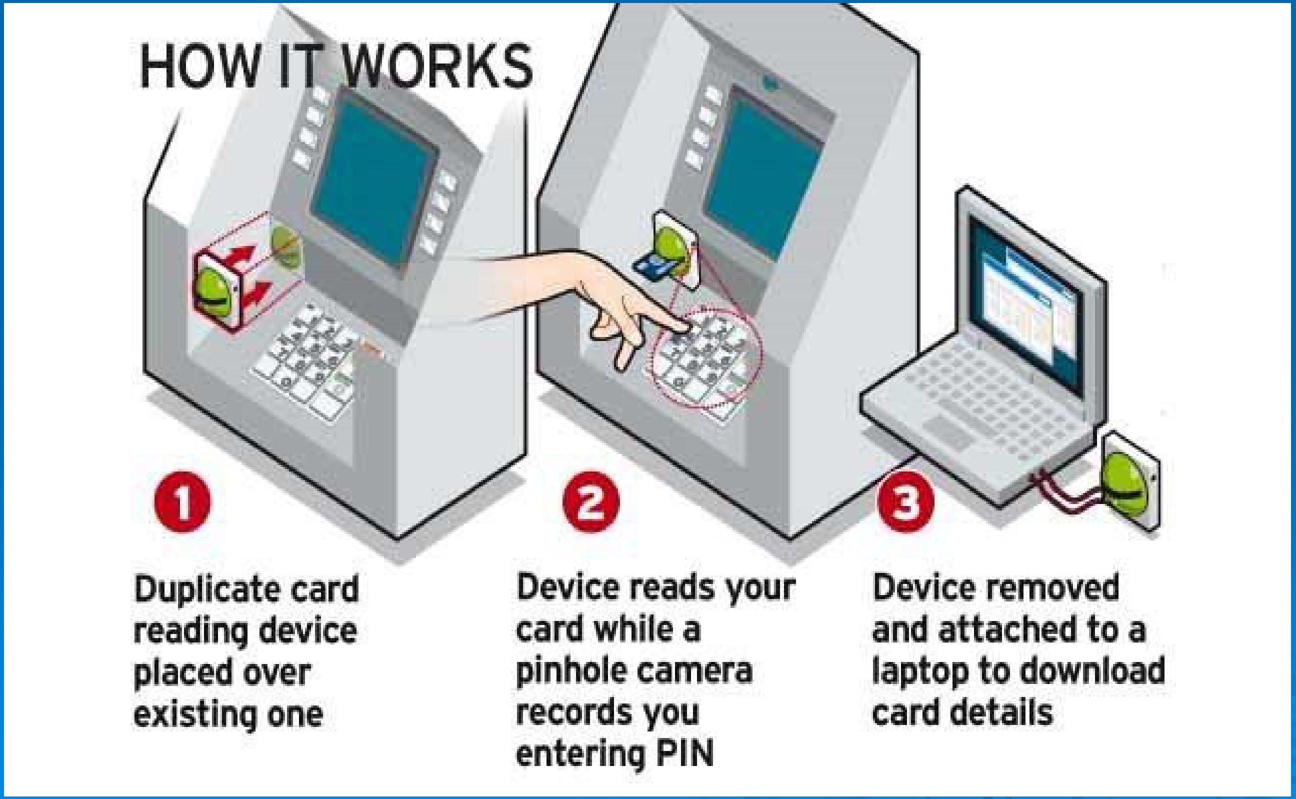
More Attacks on Passwords

- Bogus (fake) Interface
  - Attack program looks like normal login screen and records login and password of users
  - Windows has a "secure attention sequence": CTRL-ALT-DEL
  - Guaranteed to go to genuine login screen
  - A facility that guarantees a user is talking to the real system is called a **"Trusted Path"**
    - If you are security conscious, it's a good idea to always press ctrl-alt-del before you log in

- Another example:
  - Setting up a fake ATM to collect users PIN and card details
  - 'Skimming device': Reads card details

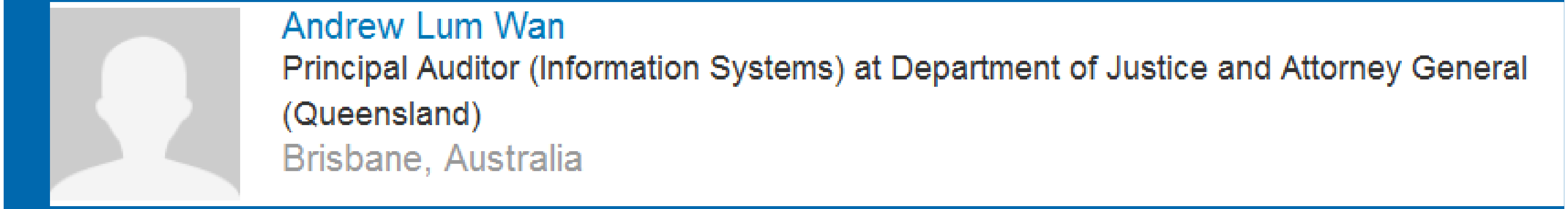
Local News Story

- **"HIGHLY sophisticated bankcard skimming devices have been found attached to ATMs in Brisbane's Queen St Mall, for the first time in Australia"**
  - http://www.news.com.au/money/banking/european-atm-skimming-machine-your-credit-cards-new-worst-enemy-in-australian-crime-first/story-e6frfmcr-1226521141277
- More recent news: http://www.cbsnews.com/news/atms-are-becoming-fatter-targets-for-bad-guys/


HOW IT WORKS
1
Duplicate card reading device placed over existing one
2
Device reads your card while a pinhole camera records you entering PIN
3
Device removed and attached to a laptop to download card details

Actually discovered by

- One of our locally known AISA members:

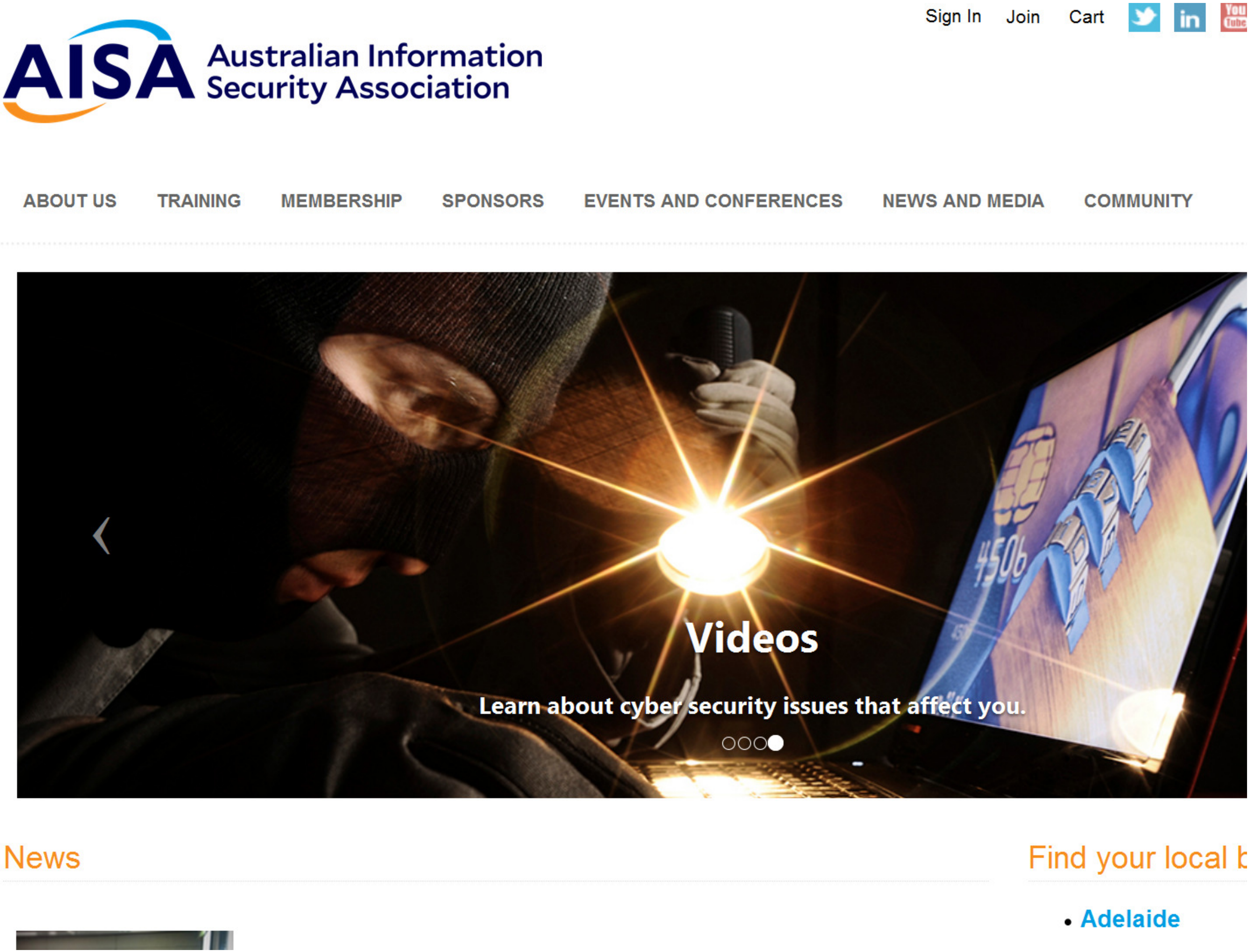


Andrew Lum Wan

Principal Auditor (Information Systems) at Department of Justice and Attorney General (Queensland)

Brisbane, Australia

- Went to use the kiosk and was suspicious
- Realised skimmer was attached and alerted authorities



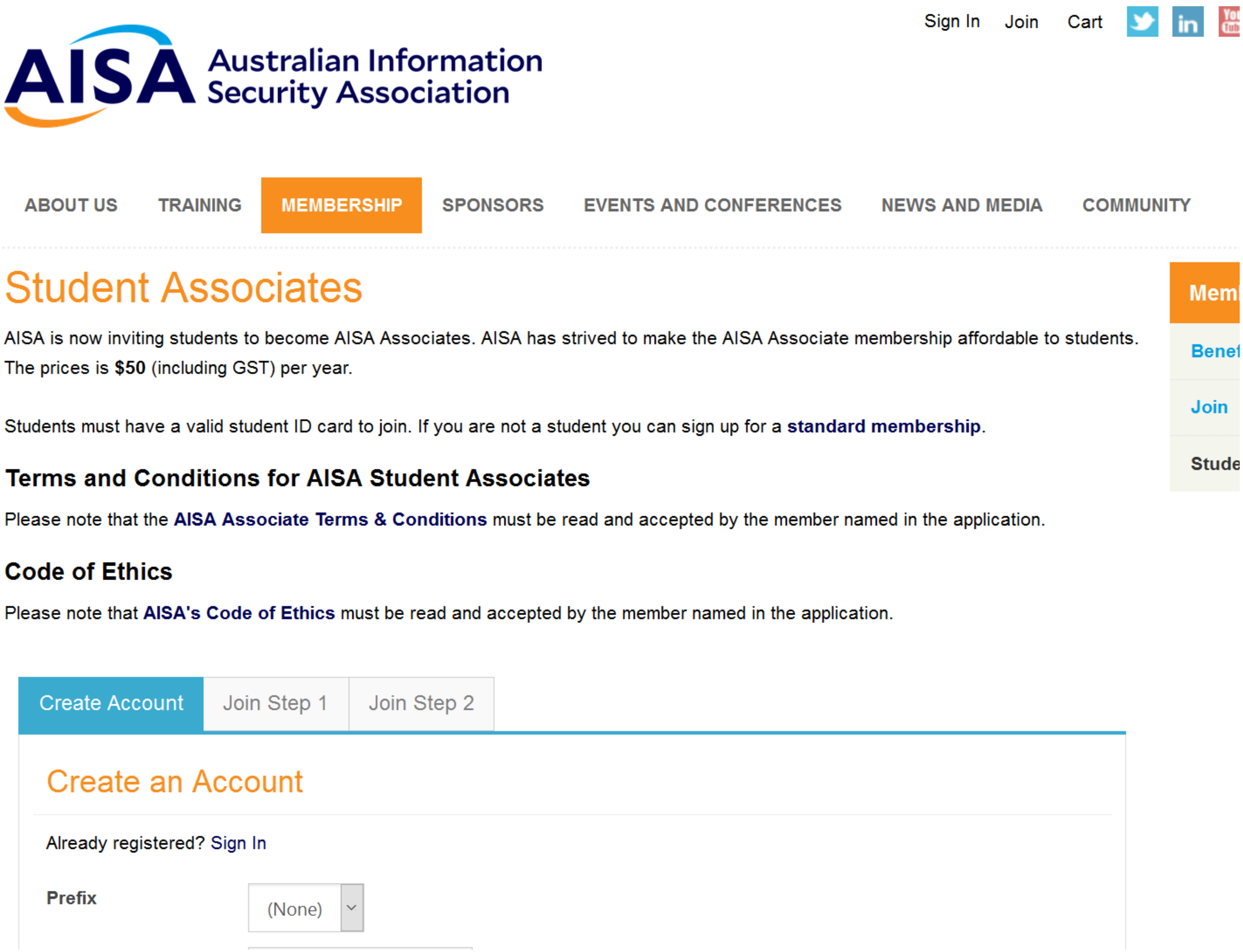
Videos

Learn about cyber security issues that affect you.

News

Find your local b

- Adelaide



Student Associates

AISA is now inviting students to become AISA Associates. AISA has strived to make the AISA Associate membership affordable to students. The prices is **$50** (including GST) per year.

Students must have a valid student ID card to join. If you are not a student you can sign up for a **standard membership**.

Terms and Conditions for AISA Student Associates

Please note that the **AISA Associate Terms & Conditions** must be read and accepted by the member named in the application.

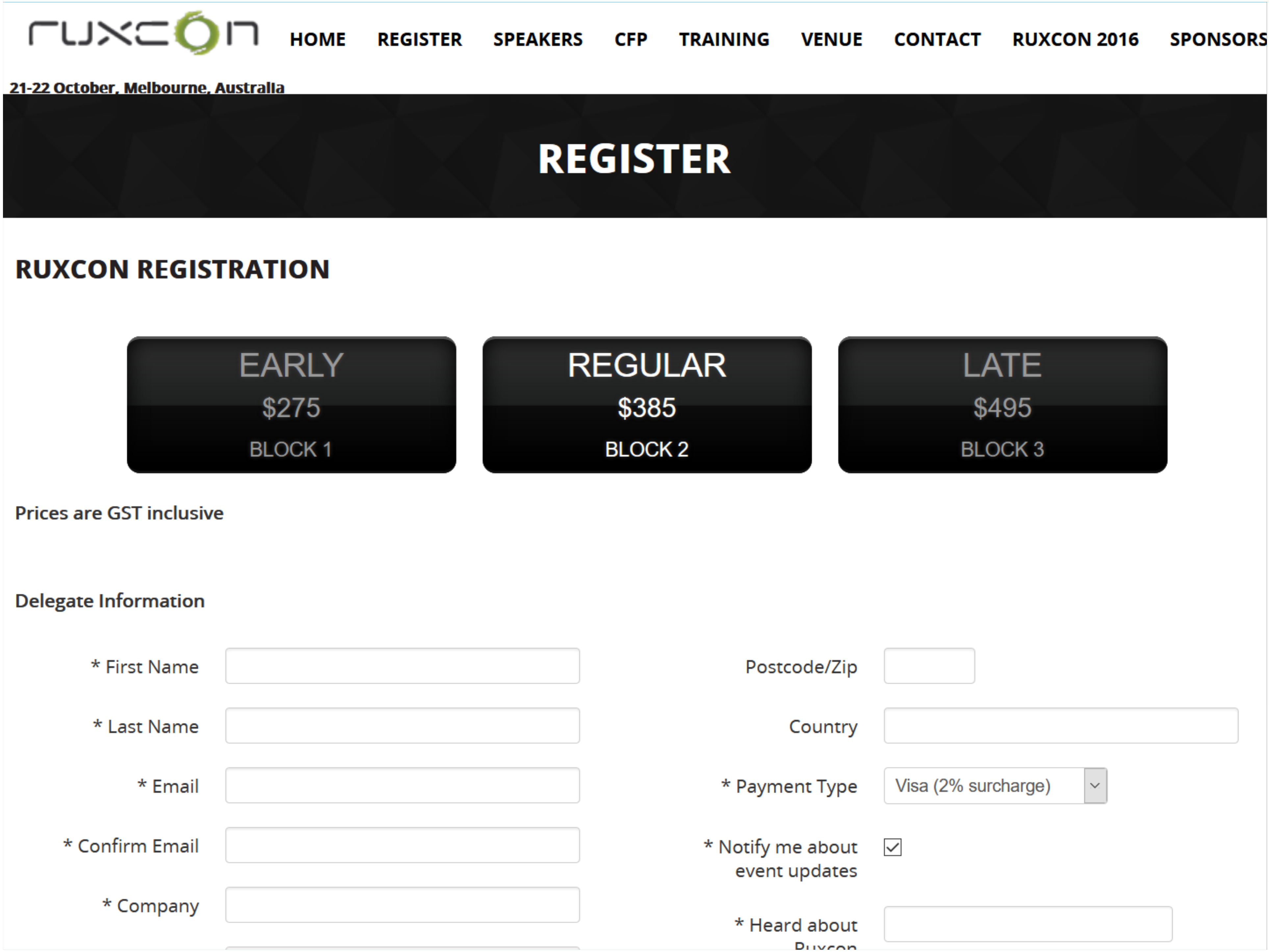
Code of Ethics

Please note that **AISA's Code of Ethics** must be read and accepted by the member named in the application.

Create an Account

Already registered? Sign In

Prefix (None)



21-22 October, Melbourne, Australia

REGISTER

RUXCON REGISTRATION

EARLY	REGULAR	LATE
$275	$385	$495
BLOCK 1	BLOCK 2	BLOCK 3

Prices are GST inclusive

Delegate Information

* First Name

* Last Name

* Email

* Confirm Email

* Company

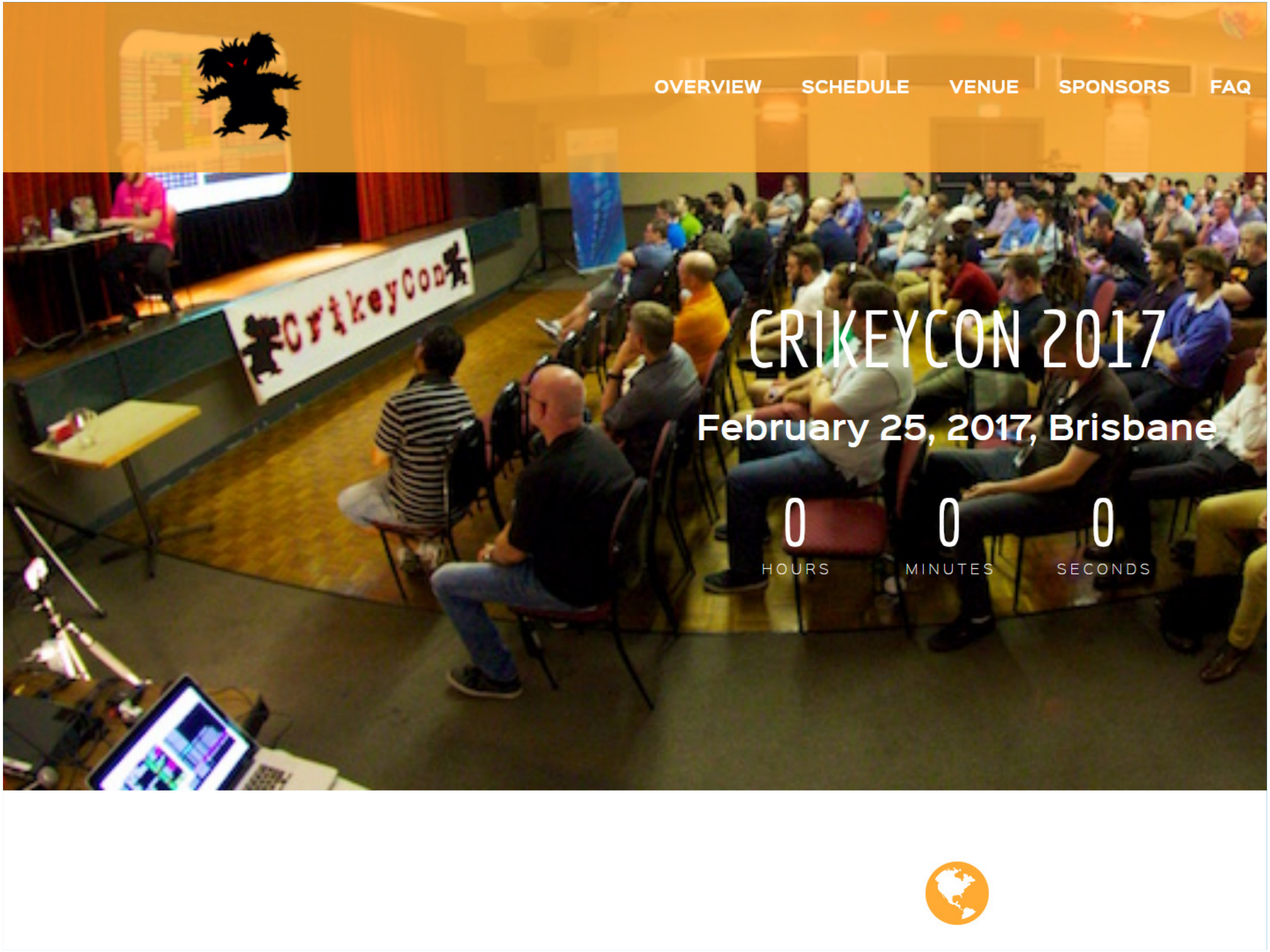
Postcode/Zip

Country

* Payment Type: Visa (2% surcharge)

* Notify me about event updates ☑

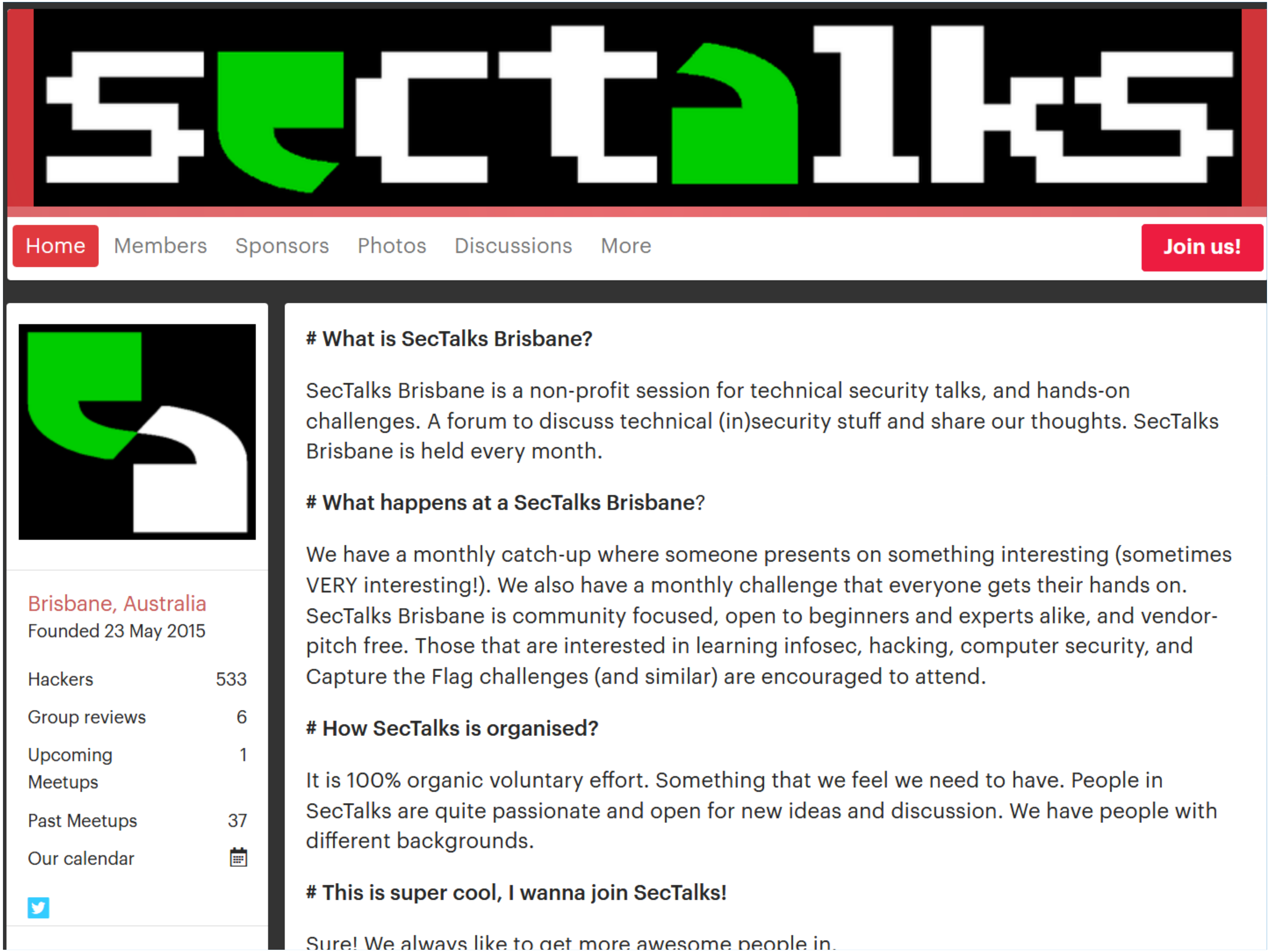
* Heard about Ruxcon



CRIKEYCON 2017

February 25, 2017, Brisbane

0 HOURS 0 MINUTES 0 SECONDS



Brisbane, Australia
Founded 23 May 2015

Hackers	533
Group reviews	6
Upcoming Meetups	1
Past Meetups	37
Our calendar	📅

# What is SecTalks Brisbane?

SecTalks Brisbane is a non-profit session for technical security talks, and hands-on challenges. A forum to discuss technical (in)security stuff and share our thoughts. SecTalks Brisbane is held every month.

# What happens at a SecTalks Brisbane?

We have a monthly catch-up where someone presents on something interesting (sometimes VERY interesting!). We also have a monthly challenge that everyone gets their hands on. SecTalks Brisbane is community focused, open to beginners and experts alike, and vendor-pitch free. Those that are interested in learning infosec, hacking, computer security, and Capture the Flag challenges (and similar) are encouraged to attend.

# How SecTalks is organised?

It is 100% organic voluntary effort. Something that we feel we need to have. People in SecTalks are quite passionate and open for new ideas and discussion. We have people with different backgrounds.

# This is super cool, I wanna join SecTalks!

Sure! We always like to get more awesome people in.

More Attacks against Passwords

- Key stroke loggers, or ‘keyloggers’
  - Record everything you type on your keyboard, including passwords
  - Can be hardware or software based
  - Can be distributed as a Malware
  - Special type of keylogger:
    - http://www.youtube.com/watch?feature=player_detailpage&v=metkEeZvHTg&t=778

- Attacks via Audit Trail
  - Systems typically log failed password attempts
  - Contain a lot of passwords, since users get login, password sequence mixed up
  - Non-existent user names like *e5gv*, *8yp* are likely to be valid passwords
  - → restrict access to log files

Attacks against Password Storage

- Problem:
  - Passwords need to be stored somewhere, needed by the system during login process
    - → vulnerability

- Originally, computer passwords were stored in plaintext, i.e. unencrypted
- Illustrative Example of the problem:
  - "In MIT's Compatible Time Sharing System, ctss (a predecessor of Multics), it once happened that one person was editing the message of the day, while another was editing the password file. Due to a software bug, the two editor temporary files got swapped, with the result that everyone who logged on was greeted with the password file."

    R. Anderson, Security Engineering

Attacks against Password Storage

- How can we solve this problem?
  - Solution by Needham and Guy
    - Store ‘encrypted’ passwords

- What is the problem with using standard encryption?
  - Passwords need to be decrypted with a secret key when users log in
  - At that stage, password (or key) is vulnerable
  - Also, how do we store the secret key?

- Solution:
  - Store a **cryptographic one-way hash** of password *p*: h(p)
  - Access to h(p) allows the OS to verify the password
  - Access to h(p) does not allow finding p

  - **cryptographic one-way hash functions** are a very important building block for a large number of cryptographic algorithms and protocols

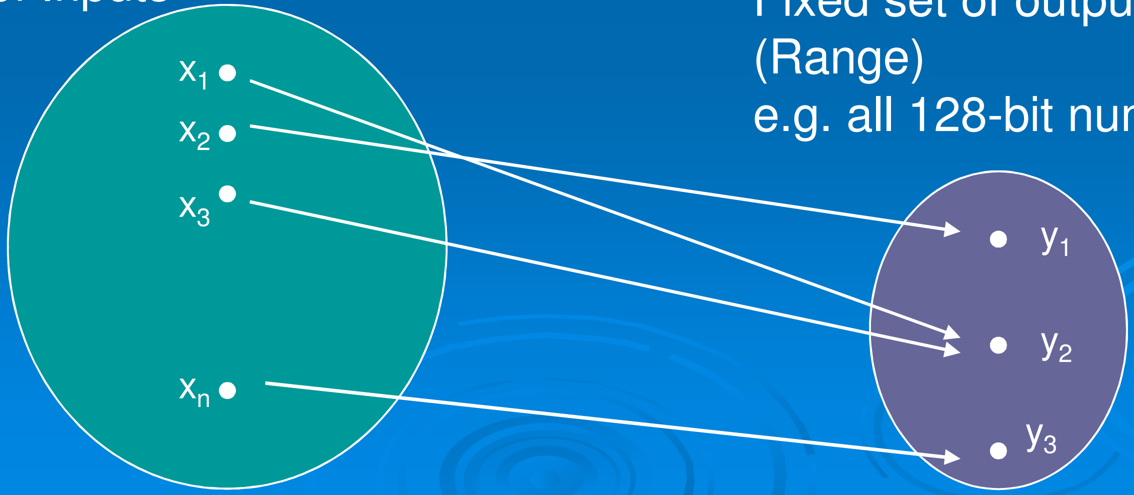
  - Let’s look at **cryptographic one-way hash functions …**

    We will come back to passwords later.

Cryptographic (One-Way) Hash Functions

- A ***hash function*** h() is a function with the following two properties
  - Compression: h() maps an input x of arbitrary finite bit length to an output h(x) of fixed, typically short, bit length
  - Ease of computation: Given h() and an input x, it is easy to compute h(x)
- A ***one-way hash function*** has the additional *one-way* property
  - For essentially all outputs y=h(x), it is 'computationally infeasible' (practically impossible) to find x
  - Also called 'pre-image resistance'

Infinite set of Inputs
(Domain)



Fixed set of outputs
(Range)
e.g. all 128-bit numbers

Cryptographic (One-Way) Hash Functions

- A ***cryptographic one-way hash function*** is a one-way hash function with the additional property of "Collision Resistance".
  - "Strong Collision Resistance"
    - It is computationally infeasible to find any two distinct inputs x_1 and x_2 so that $h(x_1) = h(x_2)$
  - "Weak collision resistance" or "2^{nd}-pre-image resistance"
    - For a given output $y=h(x_1)$, it is infeasible to find another input x_2 so that $h(x_1) = h(x2)$
      - This is harder than in the above case, therefore resistance to this is only a 'weak resistance'!
- Collision resistance is crucial for the security of algorithms and protocols, e.g. digital signatures
  - More on this later
- What does 'computationally infeasible' mean?
  - No exact definition, depends on context, it means it is not 'practical'
  - e.g. there exists no efficient (polynomial time) algorithm
  - e.g. It would take more than 1,000,000 years with current hardware

An Ideal Model of a Cryptographic One-way Hash Function

- A machine with an input and an output.
- There is an 'Elf' inside the machine, who has:
  - A very long piece of paper
  - A pen
  - An unbiased coin
- When input arrives:
  - Has this input arrived before?
    - Yes: give same output as last time (required by any function)
    - No: toss coin for new output (one toss per bit); record input and output.
- Everyone has access to this machine.
- This ideal model is called a "**Random Oracle Model**" and is often used to proof certain properties of cryptographic systems
- Important property:
  - Similar but not identical input files will most likely result in completely different hash values

Collisions - Example

- Under the Random Oracle Model, what is the probability of a collision of two different, randomly selected inputs for a hash function with a 4-bit output (n=4)?

- What is the probability that different inputs x1 and x2 have the same hash, i.e. h(x1) = h(x2)?
  - Probability of h(x2) having the same first bit as h(x1)?
    - → 0.5 (The Elf tossing a coin)
  - Probability of all 4 bits being the same?
    - → $0.5^4 = 1/16$ (Independent random experiments)

- What's the probability of a collision for a hash function with a 128 bit output?
  - $P(\text{collision}) = 0.5^n = 0.5^{128} = 2.9 * 10^{-39}$
  - ~ probability of winning 1st division price in Lotto 5 times in a row

Pre-image Attack

- ‘Work factor’ to find a ‘pre-image’, for a n-bit hash function (random oracle model)
  - i.e. for a given output h(x1) find x2 so that h(x1) = h(x2)
- Under the random oracle model, what’s the best approach to find a pre-image?
  - “brute force” attack, i.e. randomly trying different inputs.
- Inputs are randomly mapped to outputs. There are 2^n possible outputs.
- On average, how many different inputs do we have to try to find a collision?
  - on average, we can expect to find a pre-image after trying **2^n** different inputs
    - Work factor
  - This is like throwing 2^n sided dice. For n=3, we have an 8 sided dice. On average, we have to roll the dice 8 times to get a particular result, e.g. a ‘1’.
  - Due to the random mapping, randomly picking inputs is like randomly picking outputs
- For n=128 and if we use 1 billion computers that can compute 1 billion hash values a second each, how long will it take on average to find a pre-image, for h(x1), i.e. an input x2 for which h(x2) = h(x1)?
  - About 20,000 billion years

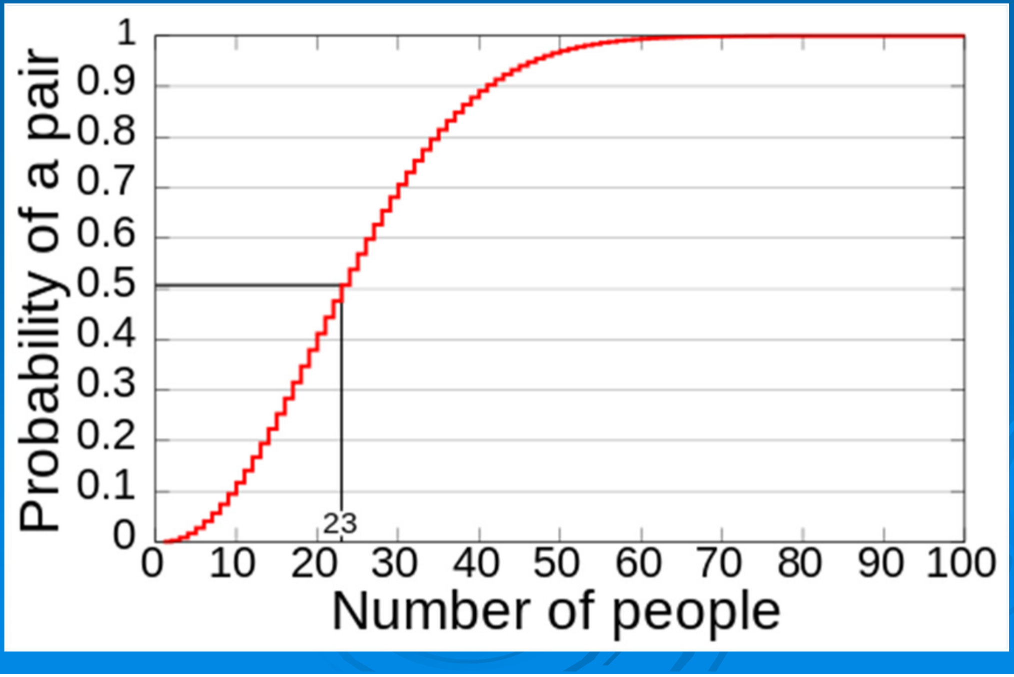
Collision Attack

- Finding a collision between any two inputs:
  - 'Collision Attack'
  - Harder or easier than pre-image attack?
    - Much, much easier than pre-image attack
  - Work factor: $2^{n/2}$ (Birthday Paradox)
  - For the example from the previous slide, the time required to find a collision is:
    - 10 seconds, compared to 20,000 Billion years!
- Example of 8 sided dice (n=3):
  - Start rolling dice, and stop if you get the same result as in any of the previous rolls.
  - Chance of collision:
    - 1st roll: 0
    - 2nd roll: 1/8
    - 3rd roll: 2/8 – (1/8 * 1/8)
    - 4th roll: …
    - …

Birthday Paradox

- What is the probability that in a class of 23 students, there is at least one with a birthday on a particular date, e.g. 1st of Jan?
  - Complementary probability: What is the probability that **no one** has a birthday on the 1st of Jan?
  - (1-1/365)^23
  - Result: 1 - (1-1/365)^23 = 0.061 → around 6%

- What's the probability that (at least) two students in a class of 23 have the same Birthday?
  - 50%
  - http://en.wikipedia.org/wiki/Birthday_problem



Cryptographic One-way Hash Functions

- Also called ‘digest function’ or ‘digital fingerprint’

- Examples:
  - md5 (message digest) – output is 128 bits.
  - SHA-1 (‘secure’ hash algorithm) – output is 160 bits.

- A cryptographic hash function is considered ‘broken’ if collisions can be found with significantly less work than brute force.

- Both md5 and SHA-1 are considered broken.

- Algorithms considered (practically) secure, so far:
- SHA-2
  - Different bit lengths, up to 512 -> SHA-512
- SHA-3
  - Announced the **Keccak Algorithm** as the winner of the 5 year NIST (National Institute for Standards and Technology) competition, October 2012
  - http://csrc.nist.gov/groups/ST/hash/sha-3/index.html
  - http://keccak.noekeon.org/

Finding Hash Collisions or Pre-images

- How can you (legally) make money finding hash collisions/Pre-images?

- Bitcoin mining
  - Finding an input with the first *n* bits of the hash output being a specific value ('proof-of-work')
  - Can adjust difficulty by changing *n*

- A lot of effort has gone into building efficient systems for calculating hashes
  - https://en.bitcoin.it/wiki/Mining_hardware_comparison
  - Example commercial bitcoin mining hardware
    - http://www.butterflylabs.com/monarch/

Possible Attacks using Hash Collisions (Pre-images)

- Most digital signature algorithms involve hashing a file before a signature is applied.
  - For performance reasons

- Let’s assume Alice has a file F1 that she wants to digitally sign. She computes h(F1) and then computes the signature
  - sig(h(F1))

- How can Bob use a pre-image attack to create a fake signature?
  - He can find a file F2 with h(F1)=h(F2)
  - He can then present F2 together with Alice’s signature sig(h(F1))=sig(h(F2)) and claim that she signed the file F2

Hash Calculator

- Can be used for tutorials, etc:
  - http://onlinemd5.com/

- How many TH/s (Terra Hashes per second) can you do? ☺