TRIESTE: A Trusted Radio Infrastructure for Enforcing SpecTrum Etiquettes

Wade Trappe

Rutgers, The State University of New Jersey
www.winlab.rutgers.edu

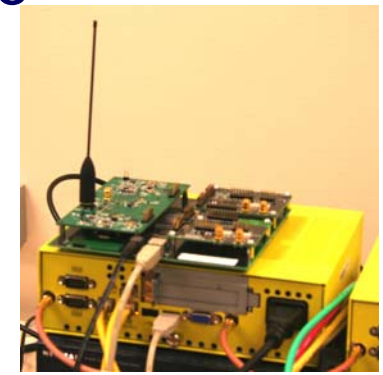
Talk Overview

- Motivation
- TRIESTE overview
- Spectrum Law Maker
  - Law/Policy Formalism
- On-board TRIESTE-TCB
  - Components
  - User request work flow
- External infrastructure
  - Distributed Spectrum Authority – Police agent
- Conclusions & Future directions

WINLAB
WIRELESS INFORMATION NETWORK LABORATORY

Cognitive Radio (CR)

- Expose the lower-layers of the protocol stack to researchers, developers and the "public"
  - scan the available spectrum
  - select from a wide range of operating frequencies
  - adjust modulation waveforms
  - perform adaptive resource allocation



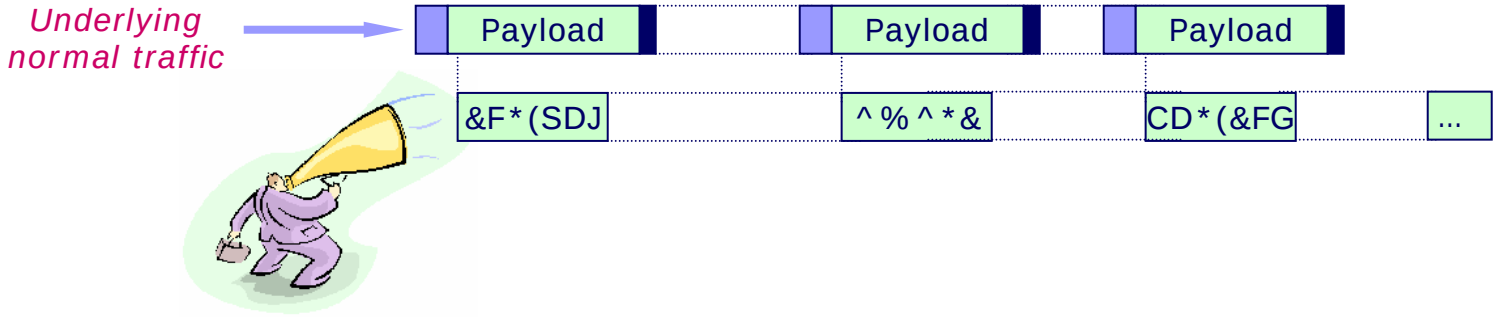
- Inexpensive and widely available cognitive radios:
  - USRP/GnuRadio – open source software support
  - Xilinx-based Rice platform
  - WINLAB-GaTech-Lucent cognitive radio platform
  - JTRS Clusters (well, not necessarily widely available…)
- An ideal platform for *abuse* since the lowest layers of the wireless protocol stack are accessible to programmers.
  - Can be reprogrammed to violate or bypass locally fair spectrum policies

Adversarial Opportunities

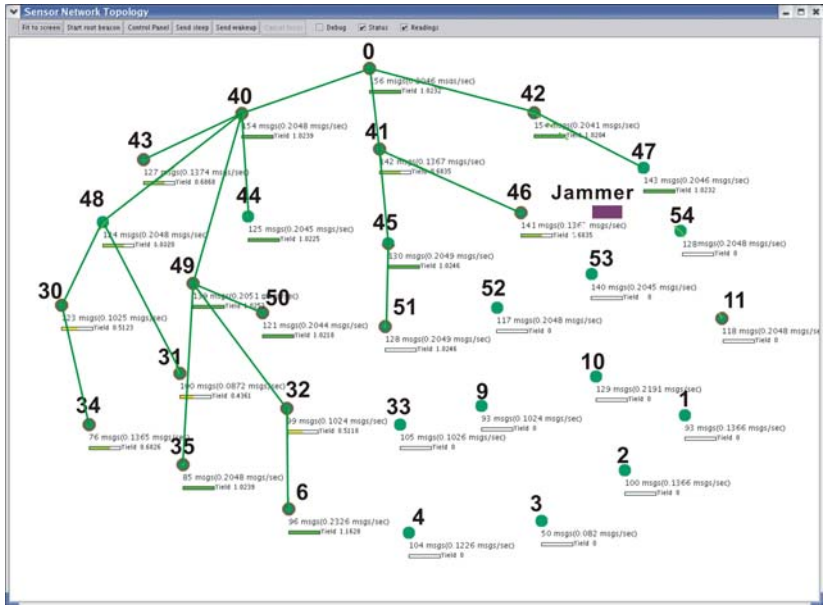
- There are many opportunities for exploitation:

1. Poor programming:
   1. CR protocols will be complex, it will be easy to write buggy implementations of etiquettes that do not achieve their goal…
   2. Runaway software processes…
2. Greedy exploitation:
   1. Decrease back-off window in an 802.11 (or comparable) implementation
   2. Ignore fairness in spectrum etiquette (many co-existence protocols assume honest participants, or honest data)
3. Simply Ignoring Etiquette
   1. Primary user returns… so-what???
4. Economic/Game-theoretic Models
   1. Standard economic models for spectrum sharing seek to support cooperation– but cooperation does not ensure trusted operation!
   2. Security is an anti-social topic!
5. Plenty more…

A Potential Attack: Radio Interference



- Many jammer models exist
- A powerful attack model: The reactive jammer
  - Stays quiet when the channel is idle, starts transmitting a radio signal as soon as it senses activity on the channel.
  - Targets the reception of a message
- What can we do?
  - Develop complicated network defense mechanisms
  - Or... Ensure that such behavior can't happen

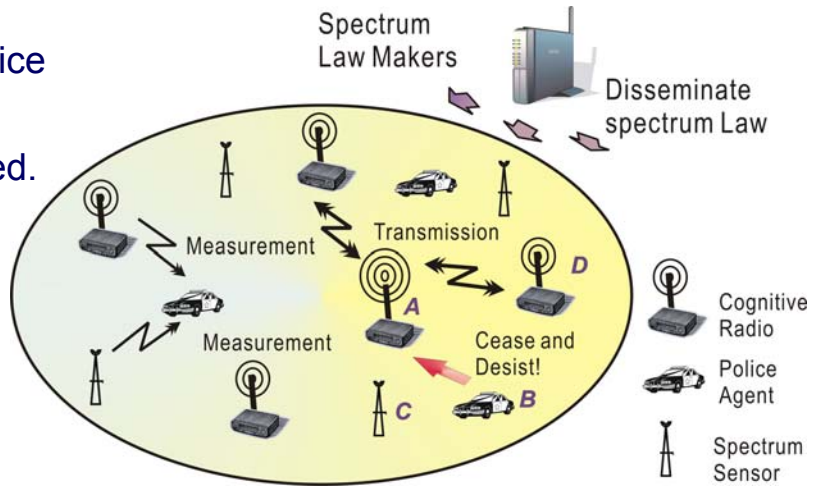


(Effect of a jammer on a network of Chipcon 1100 Radios)

WINLAB
WIRELESS INFORMATION NETWORK LABORATORY

TRIESTE – (Trusted Radio Infrastructures for Enforcing SpecTrum Etiquettes)

- Goal: to regulate the future radio environment, ensure trustworthy radio operation
- How — two complementary mechanisms
  - Motivation: Our normal day-to-day society…
  - On-board enforcement – restricting any violation attempt from accessing the radio:
    - running its own suite of spectrum etiquette protocols
    - behaves according to acceptable communal policies
  - An external monitor infrastructure:
    - Distributed Spectrum Authority (DSA) — police agent observes the radio environment
    - DSA will punish CRs if violations are detected.



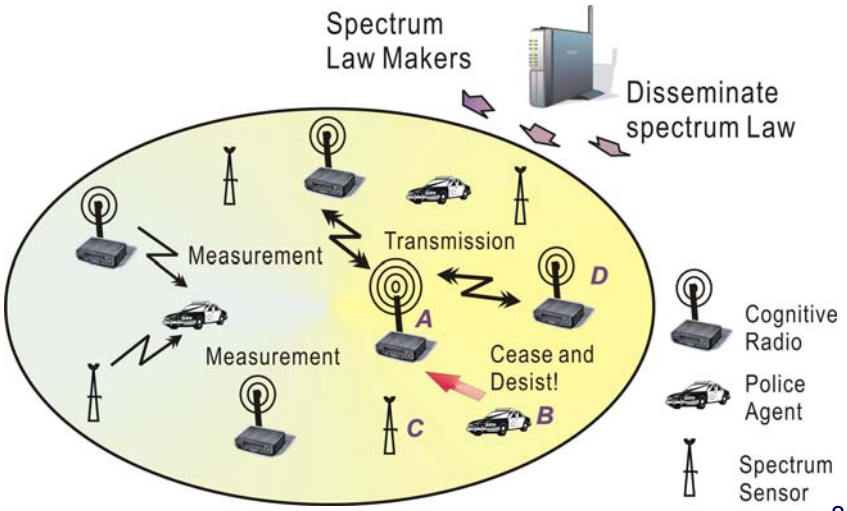
WINLAB
WIRELESS INFORMATION NETWORK LABORATORY

TRIESTE
– Spectrum Law Maker

Spectrum Law Maker

- Defining the spectrum laws,
  - *Laws:* restrict the spectrum etiquette policies that are programmed by CR users
    - Example: an entity should not leak energy outside the spectrum it has negotiated
  - *Spectrum etiquette policies:* defined by individual cognitive radio users, or spectrum owners, and they have to obey the spectrum laws.
    - Example: entity can continuously access the allocated spectrum during a specified time period

- Disseminating spectrum laws to the DSA and cognitive radios
  - ==> The law should be defined using *a formal language*, a format that is understandable by the radio device

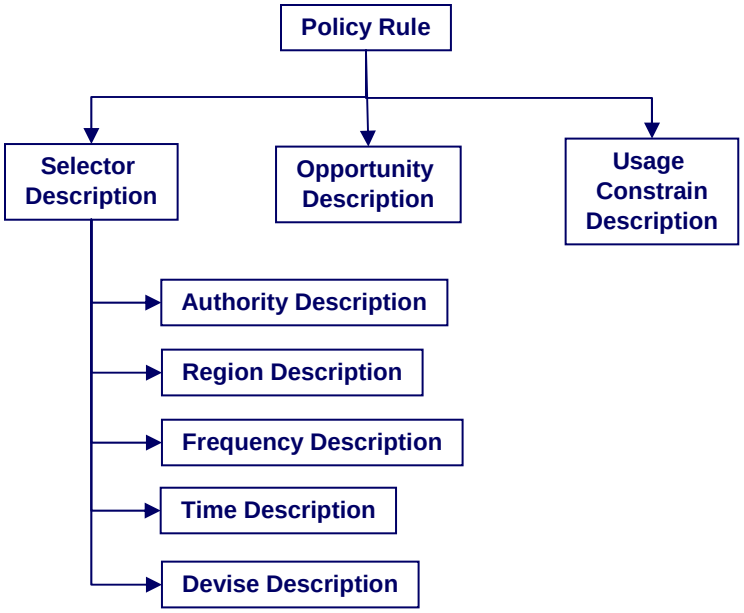


Spectrum Law/Policy Formalism

- Traditional way:
  - laws/policies are published in a human readable form => interpreted by humans => hard-coded into radio device
  - inefficient, inflexible and non-scale

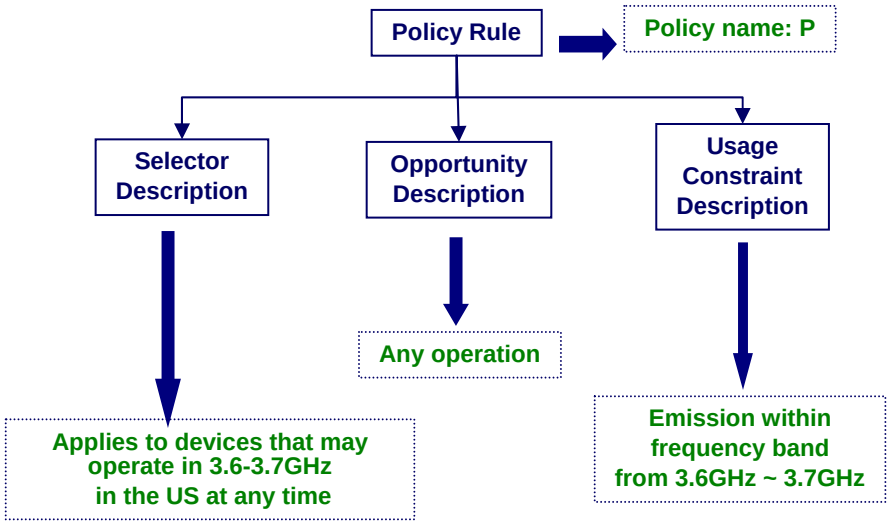
- XGPL (XG Policy Language)
  - written in XML (Extensible Markup language)
  - a declarative language based on *facts and rules* instead of a procedural language

- Policy Rule
  - Selector Description:
    - used to filter policy rules, for a given situation
  - Opportunity Description
  - Usage Constraint Description



Spectrum Law/Policy Formalism – example

- Spectrum access policy:
  - This policy applies to CR devices that are capable of operating in 3.6-3.7GHz; Transmissions shall be contained within 3.6*GHz* to 3.7*GHz*



Access Control and Spectrum Law Logics

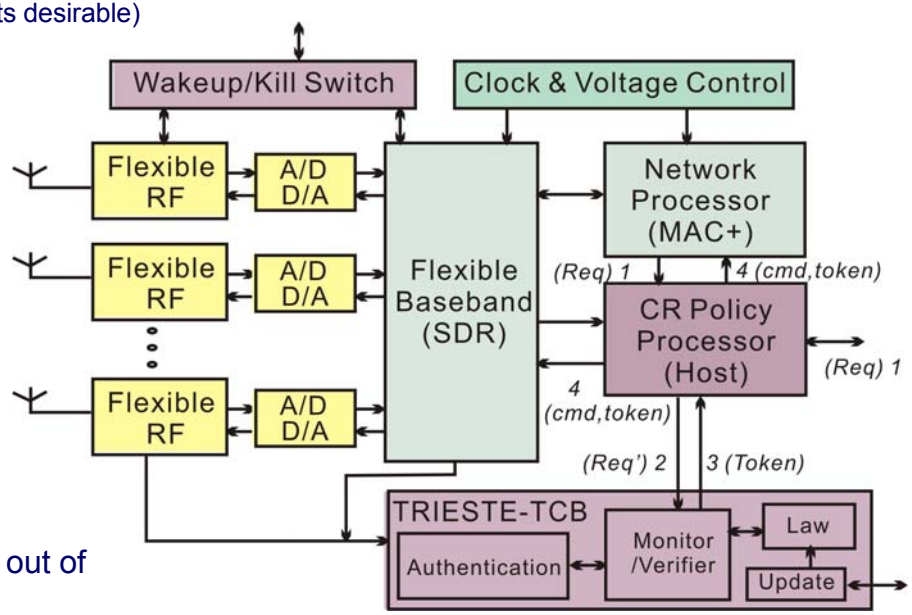
- We are proposing that explicit forms of on-board access control be implemented
  - Access control requires logic checking
  - Most basic level are access control lists
    - E.g. Device/Type X is allowed (or not) to perform action Y
  - This is too simple for most CR scenarios
- In order to allow for more complex scenarios, AC must use policy specifications that allow for complex actions
  - Example: Take/Grant models that allow for (grant) delegation of privileges when a primary user is absent, and (take) privileges when a user returns
- Policy specifications may need to use multiple information types
  - On-board timers, spectrum sensing, certificates and credentials signed by an authority
- Formal logic tools can be used to implement CR-AC policies
- Potential Starting Points:

1. XGPL
2. J. Halpern and R. van der Meyden. A logic for SDSI's linked local name spaces. *Journal of Computer Security* 9:47−74, 2001.
3. LGI

TRIESTE – On-board TRIESTE-TCB

TRIESTE-TCB

- What is the TRIESTE-TCB (Trusted Computing Base)
  - A virtual block includes all the hardware and software that enforces universal laws and etiquette policies
  - A controlled gate that users have to go through to access radio
    - (standard tamperproof components desirable)



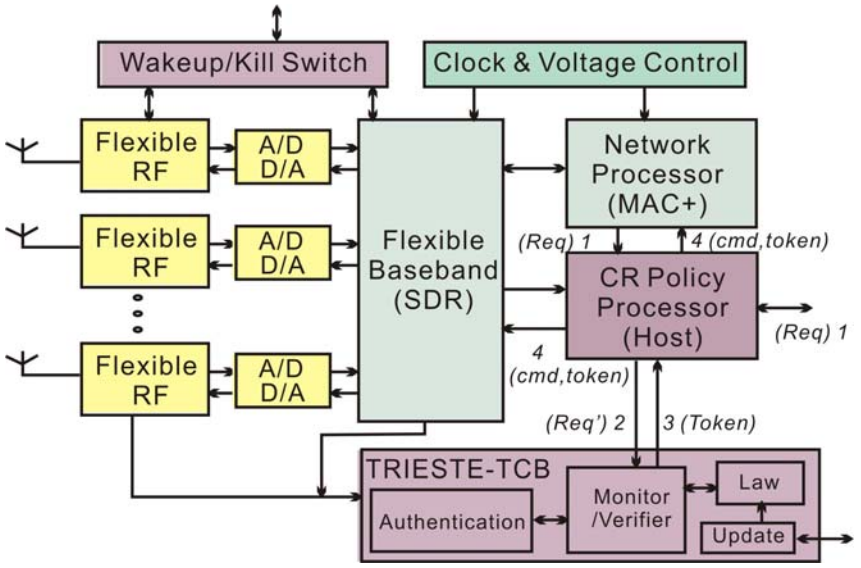
- Components:
  - ***CR processor*****:** programmable by the User; performs request filtering based on user defined spectrum etiquette policies
  - ***Monitor/Verifier:*** a *Controller* which can interpret and enforce any well-formed *Law.* Verify user's radio access request, monitor the on-board radio activity.
  - ***Wake up/Kill Switch***:
    - "wakeup": brings the baseband processor out of a deep (low power) sleep.
    - "kill": stops the corresponding ongoing radio activities.
  - ***Update:*** allows the laws evolve over time, accepts a new law only if it is signed by the regulating authority,

User request work flow

- User request and user's credentials (req, user) =>CR policy processor
- If (req,user) is valid based on Spectrum etiquette policies,
  - (req, user) => Monitor/Verifier.
  - Otherwise, either modified (req', user) are sent, or no (req) is sent.
- Monitor/Verifier checks (req', user) against Spectrum access Laws.
- If (req', user) is valid,

⇒ issues a privilege token *(spec-access-details || timestamp || hash())*

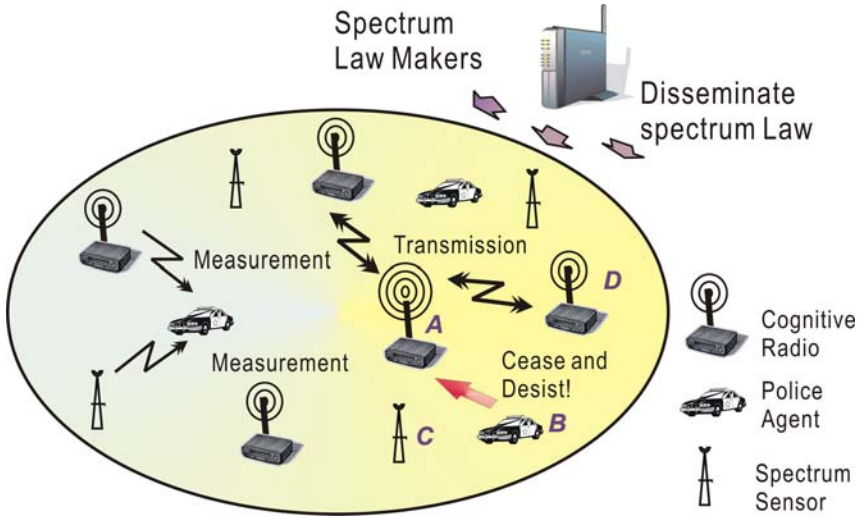
- With the valid token, the radio access is granted.



TRIESTE – External infrastructure

External infrastructure

- Spectrum sensors: monitors the local radio environment
  - How much data to collect and distribute?
- Distributed Spectrum Authority (DSA) – Police agent
  - Collecting radio measurements from *cognitive radios* & *auxiliary spectrum sensors*.
  - *Discovering the Crime* based on the geographically distributed radio measurements
  - *Punishing* the individual radio device that violate the spectrum law/policy (Authenticated Kill Switch)
  - Localizing the misbehaving device and seizing it.



External infrastructure – issues

- *Discovering the Crime*: radio measurements can be supplied by potentially greedy/rogue users. How to correctly detect violation based on potentially biased data?

- *Identification*: identify individuals/CRs associated with a crime
  - measure transmitter radio signatures based on unavoidable and random fabrication difference (see Signal Prints and NIST studies)
  - embed hard-to-alter (or mandatory) RF fingerprints within every transmission

- *Punishment policies formalize:* XGPL is NOT designed to specify what to do if spectrum abuse is detected
  - Extend XG policy language to include punishment definition.
  - If the punishment rule is selected and activated, then new punishing rules with certain expiration period will be generated based on the level and type of punishment, and be inserted into the existing spectrum polices for certain amount of time.

Discussion

- As Cognitive Radios (CRs) become ubiquitous in the future there will be attempts to misuse the highly open and granular control provided to the radio interface.
- We proposed a framework TRIESTE to ensure that radio devices are only able to access/use the spectrum in a manner that conforms to their privileges.
- Two levels of etiquette/law enforcement mechanisms:
  - on-board trusted computing base/module (TCB)
  - an infrastructure external to individual CR
- There are many sub-problems to be tackled…
  - It will take a village!

Future work

- Evaluate the impact of using an initial TRIESTE-TCB on the performance of a CR.
  - Map out the interplay between policies, their interpretations, and their enforcement using onboard mechanisms.
- Identification mechanisms to recognize CRs.
  - Introducing RF signatures into a CR's transmissions
- Correctly detect violation based on potentially biased data
  - Specification-based anomaly detection can be slow
- Integrate localization and identification system, using spectrum sensor readings and the cooperation of neighboring CRs.
- Access Control and Punishment policy formalization
  - Should require formal logics…
- Implementation
  - What amount of CAs can be implemented on-board?
  - Initial starting point: "pretend" we have a tamperproof component through a virtual partition (e.g. within Micro-blaze soft-core on the WINLAB-Xilinx Cognitive Radios)