

# Genetic Algorithms and Network Intrusion Detection

Mark McFadden – MBI 640





# Presentation Outline

- What is Network Intrusion Detection System (IDS)?
- Genetic Algorithms (GAs) and IDS.
- GAs 101 - how to form a GA via the IDS Problem domain
- Science Class Review
- Creating the IDS “Chromosome”
- The “DNA” of a Genetic Algorithm
- Intrusion Detection System GA Demo
- Future Steps
- Questions/Discussion



# Network Intrusion Detection

- The goal of intrusion detection is to identify entities attempting to subvert in-place security controls.<sup>1</sup>

# Denial of Service (DoS) Attack

- Imagine that an intruder wanted to make a telephone system unusable by telephone customers. How would they do this? One way would be to make call after call in an attempt to make all circuits busy. This type of attack is called a denial of service, or DoS, attack.<sup>2</sup>





## A Genetic What?

- A Genetic Algorithm (GA) is a family of computational models based on principles of evolution and natural selection.<sup>3</sup>
- A GA employs a set of adaptive processes that mimic the concept of “survival of the fittest.”



# How will this help Intrusion Detection?

- A network IDS, using either a network tap, span port, hub, or firewall, collects traffic based information that traverse a given network.<sup>3</sup>
- This traffic data is then used by the GA for the creation a set of rules for an Intrusion Prevention Rule based system.
- Intrusion prevention follows the same process of gathering and identifying data and behavior, with the added ability to block or prevent the activity.



# Genetic Algorithms: 101

- The definition of the word genome is “the haploid set of chromosomes of an organism”<sup>5</sup>
- So, first, we convert the problem in a specific domain into a model by using a chromosome-like data structure.<sup>3</sup>



Problem  
Domain → Chromosome

- For us, since we want a more “evolved rule set” for intrusion detection, we will convert a potential intrusion (problem domain) into a chromosome.



# Getting the Suspected Intrusion

- Here is the typical content of a firewall log entry:
  - Time | Action | Firewall | Interface | Product | Source | Source Port | Destination | Service | Protocol | Translation | Rule
- For our purposes we will look at:
  - Source IP address | Destination IP address | Destination Port | Protocol | Bytes Sent by Originator | Bytes sent by Responder.



# Customizing the Suspected Intrusion

- Next we note a suspected attack with the following record:

*Source IP: 192.19.54.155 | Target IP: 109.1.1.20 (a database server) | Destination port: 8184 (is for internal data port) | Protocol: File Transfer Protocol | Originator sent 7,500 bytes of data | The responder (database) sent 2,500,000 bytes of data*



# Suspect Record → Chromosome

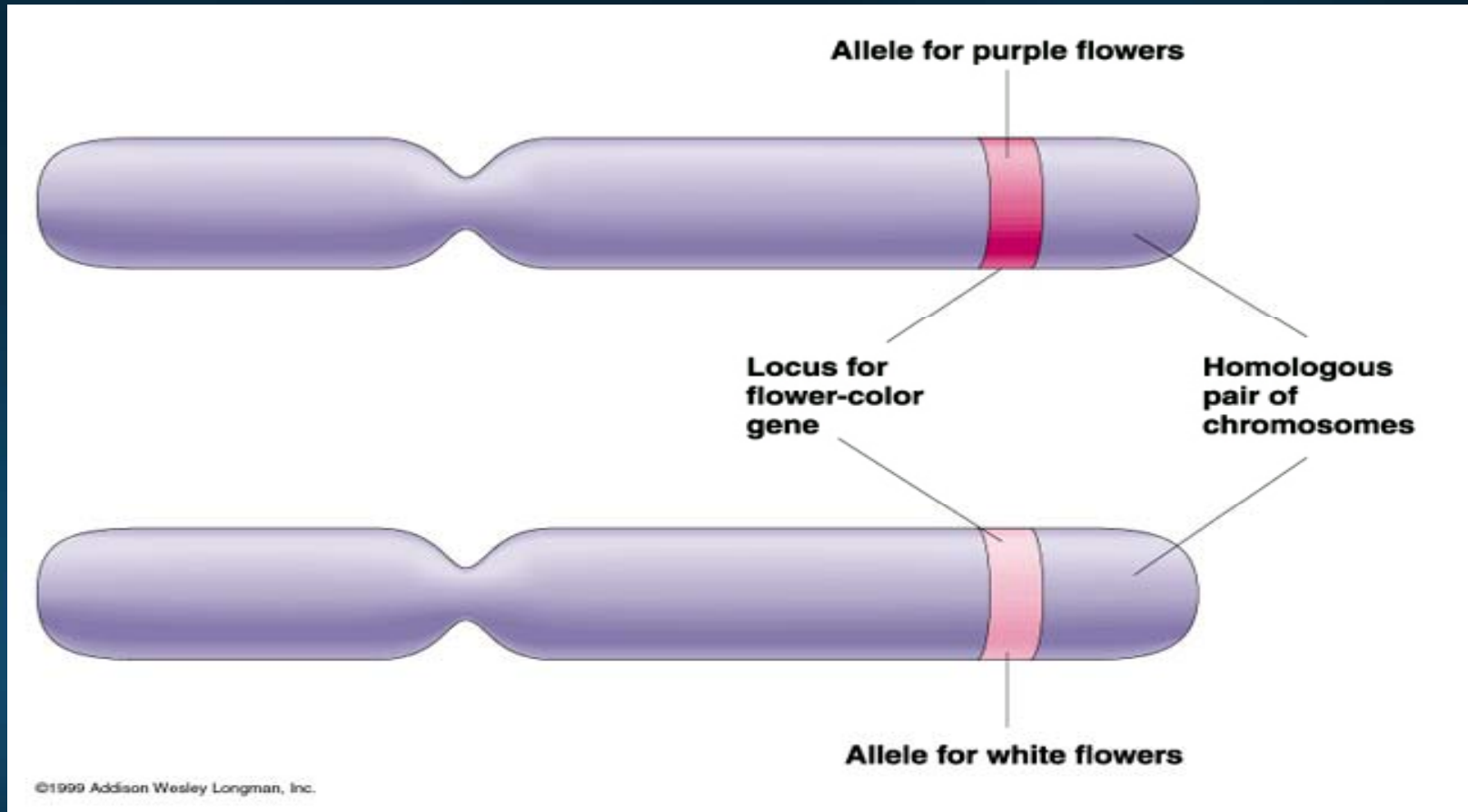
- Now that we have a log record that is suspect, we can create the IDS Chromosome.
- In order to use a Genetic Algorithm we need to make the problem domain into a “Chromosome.”



## Review: Remember Science Class?

- Chromosome – The self-replicating genetic structure of cells containing the cellular DNA that bears in its nucleotide sequence the linear array of genes.<sup>7</sup>
- Allele – One of the variant forms of a gene at a particular locus, or location, on a chromosome.<sup>8</sup>

# Review: Chromosome/Allele



From: <http://www.csulb.edu/~kmacd/361-6-Ch2.htm>



# Creating Our IDS Chromosome - I

- Within a rule based system, the rules stored in the rule base are usually in the following form:  
if <condition> then <action><sup>6</sup>
- In our case:
  - if {the connection has following information: source IP 209.11.1.155; destination IP address: 109.1.1.17 ~ 109.1.1.21; destination port number: 8184; the protocol used is FTP; the originator sent more than 10,000 bytes of data; and the responder sent more than 250,000 bytes of data } then {stop the connection}<sup>3</sup>



# Creating Our IDS Chromosome - II

- Next, we consider the rule set and place it into a tabular format for clarity.

Attribute	Range of Values	Example Values	Descriptions
Source IP address	0.0.0.0 ~ 255.255.255.255	125.19.54.155	122.19.54.155 is a suspect IP address.
Destination IP address	0.0.0.0 ~ 255.255.255.255	109.1.1.20	IP Address 109.1.1.17 ~ 109.1.1.21 are database servers.
Destination Port Number	0 ~ 65535	8184	Destination port number, indicates this is a http service. 8184 is for internal data access.
Protocol	1 ~ 20	5	The protocol for this connection FTP. (5 = FTP)
Number of Bytes Sent by Originator	0 ~ 250 KB	10.5 KB	The originator sends 10,500 bytes of data
Number of Bytes sent by Responder	0 ~ 1 MB	2.5 MB	The responders sends 2,500,000 bytes of data

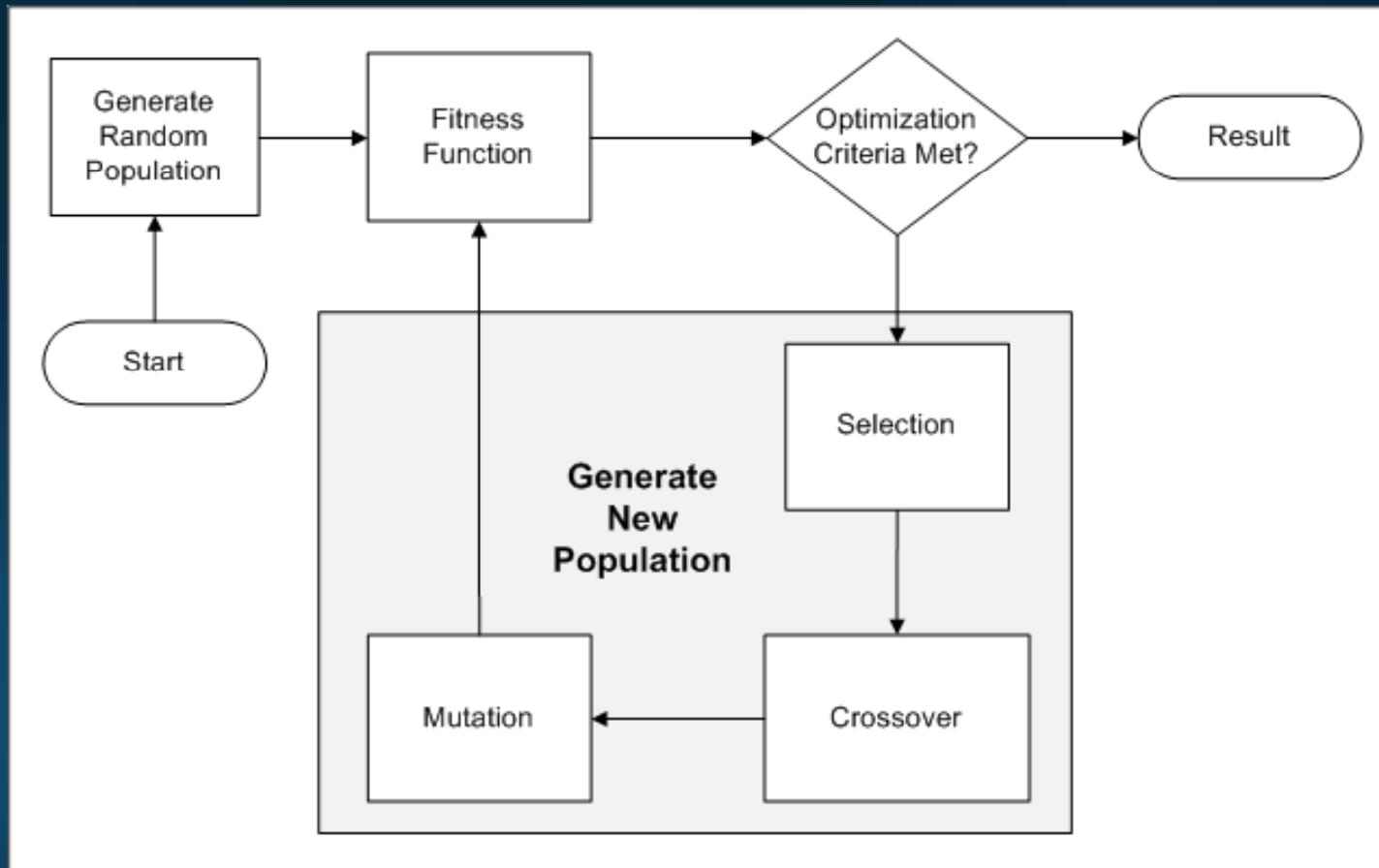


# Creating Our IDS Chromosome - III

- We can convert the tabular rule set into a chromosome form, with each item in the left column representing an allele of the chromosome with its corresponding value in the right column.

<b>Source IP address</b>	125.19.54.155 converted to 2098411163
<b>Destination IP address</b>	109.1.1.20 converted to 1828782356
<b>Destination Port Number</b>	8184
<b>Protocol</b>	5
<b>Number of Bytes Sent by Originator</b>	10500
<b>Number of Bytes sent by Responder</b>	2500000

# What is the “DNA” of a GA?



Adapted from:

Pohlheim, Hartmut. (2003). *Genetic and Evolutionary Algorithms: Principles, Methods and Algorithms*.

Genetic and Evolutionary Algorithm Toolbox. From: <http://www.geatbx.com/docu/alginde-01.html>



# Fitness Function - Step 1

- First, the general outcome is determined based on whether a “gene” (or allele) matches the pre-classified data set that was obtained from a network device such as a firewall log. Then multiply the weight of that field to the “matched” value that is either 1 or 0<sup>3</sup>.

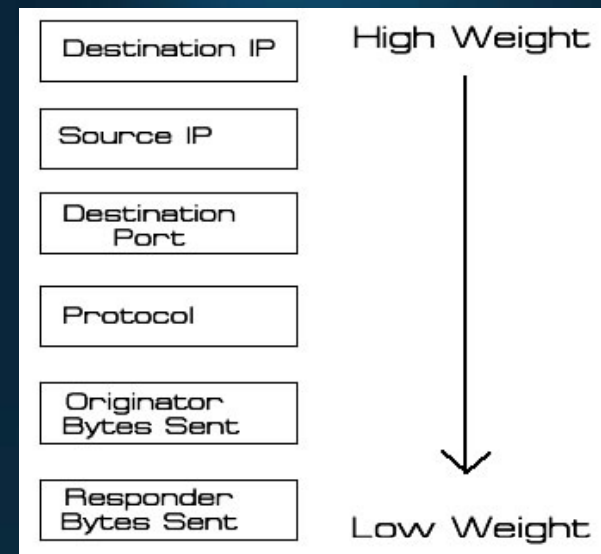
$$\text{Outcome} = \sum_{i=1}^6 \text{Matched} * \text{Weight}_i$$

# Fitness Function - Step 1

## - continued

- Weight values are applied to the different genes as historically reported by network devices. All genes denoting the Destination IP address have the same weight value<sup>3</sup>.

$$\text{Outcome} = \sum_{i=1}^6 \text{Matched} * \text{Weight}_i$$





## Fitness Function - Step 2

- The delta value or absolute difference between the “outcome” of the chromosome and the suspicion\_level is then computed using the following equation. The suspicion\_level is a value that indicates if the historical gene value and the suspicious gene value are considered a “match.” The actual value of suspicion\_level also reflects observations from historical data<sup>3</sup>.

$$\Delta = | \text{outcome} - \text{suspicion\_level} |$$



## Fitness Function - Step 3

- If the delta level is high enough, a penalty value is calculated using this delta (or absolute difference). The “ranking” in the equation below indicates whether or not a network intrusion is easy to establish. If the ranking value is high, the historical data should verify the determination<sup>3</sup>.

$$\text{penalty} = \left( \frac{\Delta^* \text{ ranking}}{100} \right)$$



## Fitness Function - Step 4

- Finally, the chromosome's fitness is then computed using the above penalty. The scope of the fitness result is between 0 and  $1^3$ .

$$\text{fitness} = 1 - \text{penalty}$$

- Following the running of the fitness function the fitness level is reviewed. If the fitness level is not obtained, the algorithm then evolves through the selection, crossover (recombination), and mutation functions



# Selection

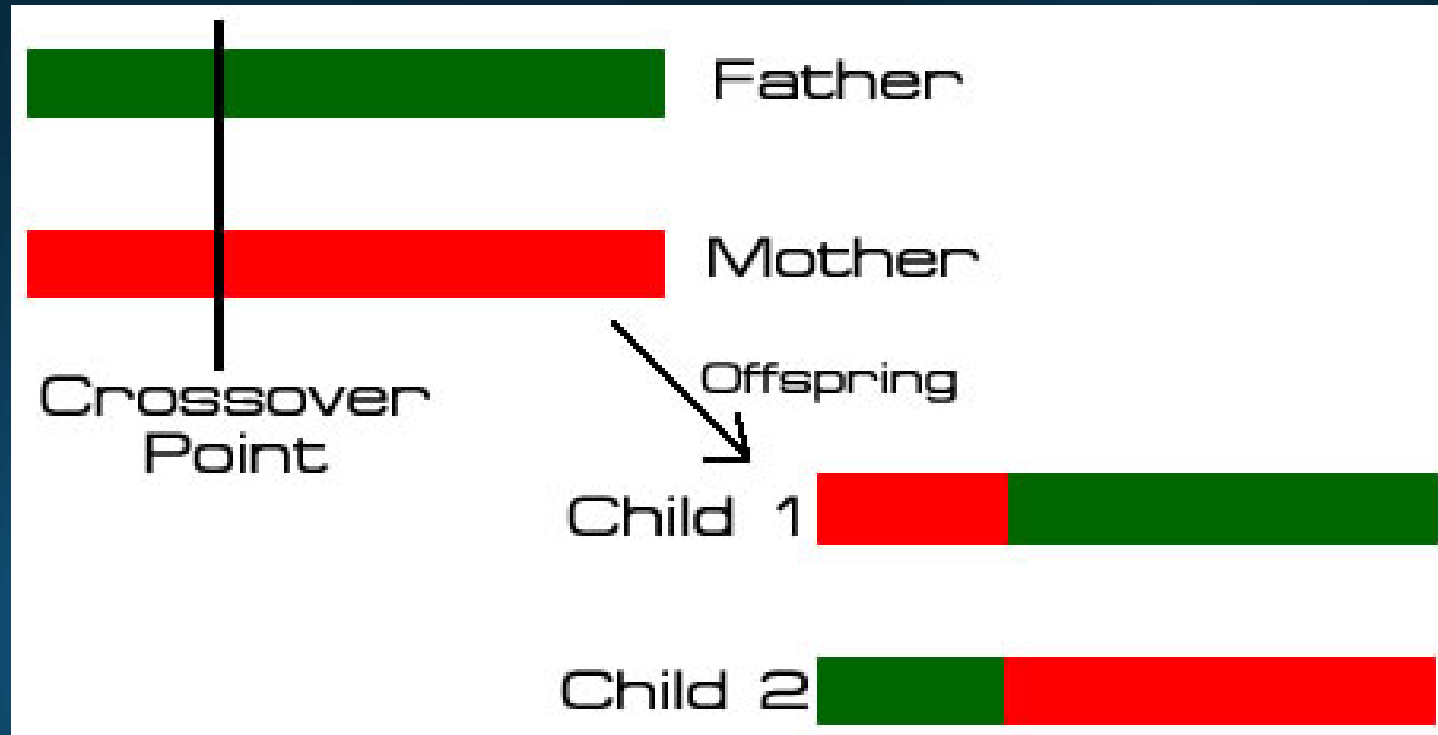
- In essence, the GA begins to “experiment” with the existing set of chromosomes by combining and refining the genes contained within each chromosome. The objective is to produce new chromosomes that form a new generation of possible solutions to evaluate.<sup>9</sup>



# Crossover

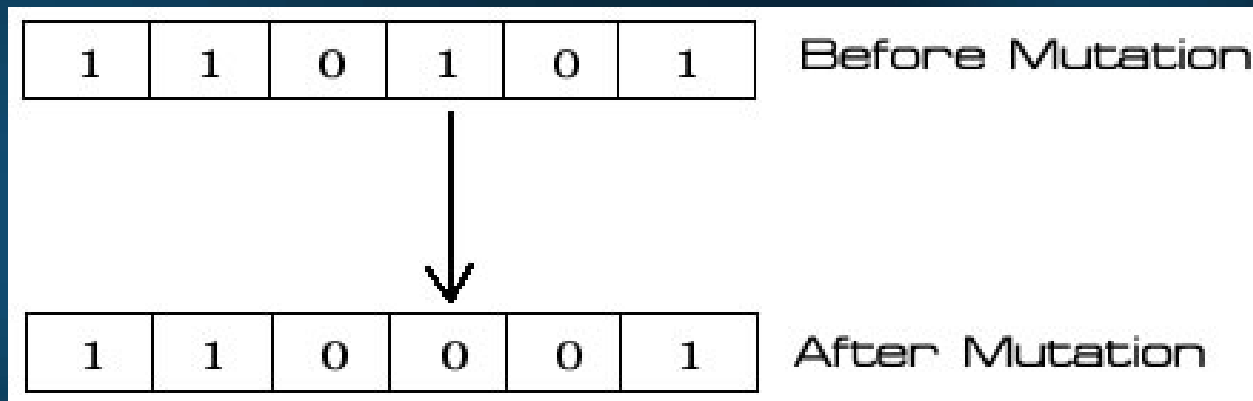
- The crossover operation allows the GA to create a new chromosomes that share positive characteristics while simultaneously reducing the prevalence of negative characteristics in an otherwise reasonably fit solution.<sup>10</sup>

# Crossover (continued)



# Mutation

- The final step in the refinement process is mutation. The mutation phase randomly changes the value of a gene from its current setting to a completely different one.<sup>10</sup>





# Evolved Outcome

Convert this:

Source IP	Destination IP	Destination Port	Protocol	Originator Bytes	Responder Bytes
2006384639	1996554516	8184	5	10500	2500000

To This:

*if* {the connection has following information: source IP 125.19.54.155; destination IP address: 119.1.1.17 ~ 119.1.1.21; destination port number: 8184; the protocol used is FTP; the originator sent more than 10,000 bytes of data; and the responder sent more than 250,000 bytes of data }  
*then* {log the intrusion and stop the connection}



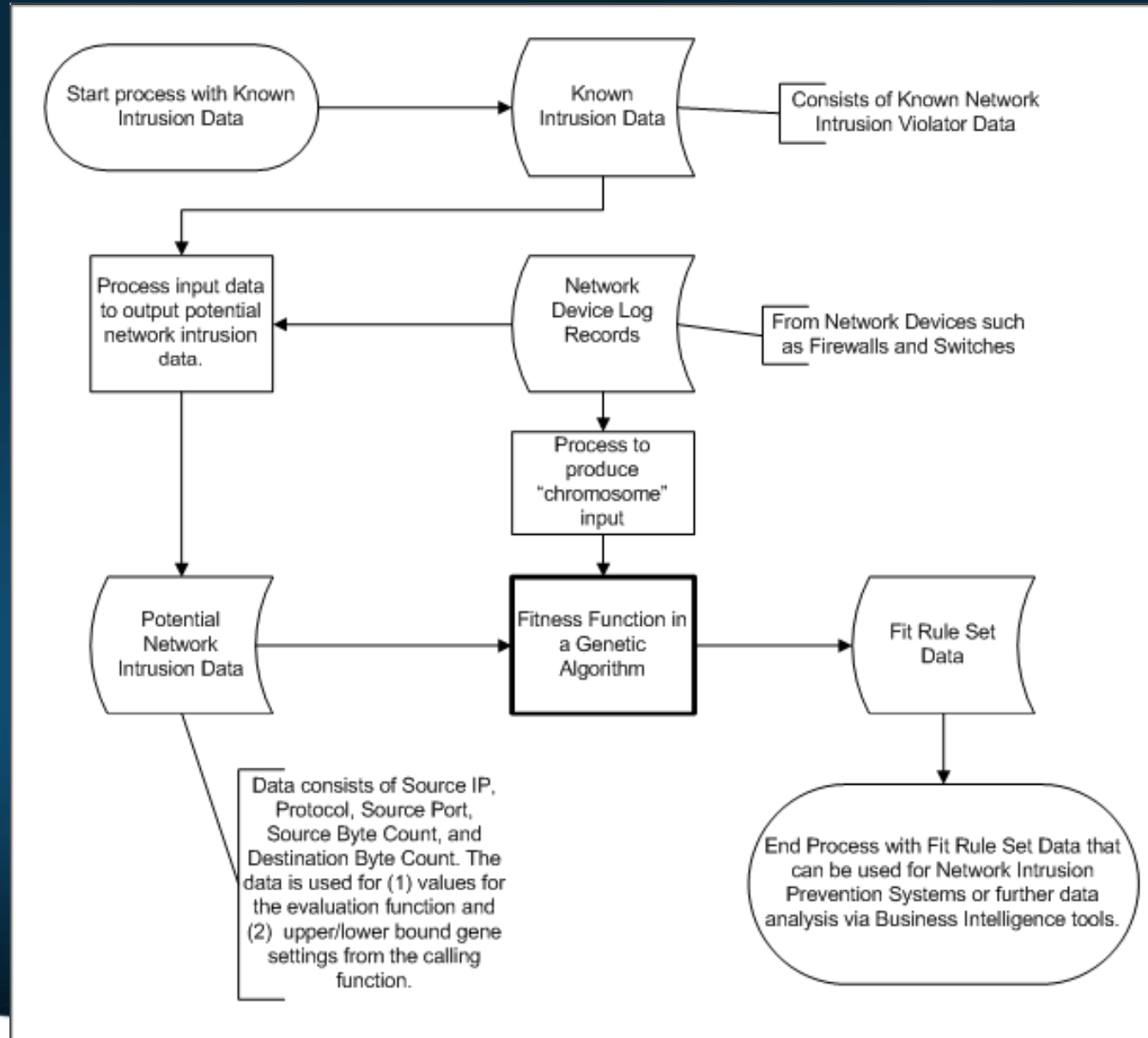
# Genetic Algorithm Demo<sup>11</sup>



# Future Steps

- Apply an actual anomaly based data and network device data to form the input “chromosomes”, the gene range values, and the values for the evaluation function.

# Future Steps-continued





# Future Steps-continued

- Compare GA results with existing anomaly rule set for effectiveness.
- Data mine the GA results for patterns or data clusters in the output and then analyze for discoveries.
- Utilize Genetic Programming, which enhance GAs since they produce dynamic programs instead of static chromosomes, to produce more multifaceted and flexible outcomes.



# References

1. Berge, Matthew. "Intrusion Detection FAQ: What is Intrusion Detection?" From [http://www.sans.org/resources/idfaq/what\\_is\\_id.php](http://www.sans.org/resources/idfaq/what_is_id.php). Retrieved 2-29-08.
2. CERT. From [http://www.cert.org/tech\\_tips/denial\\_of\\_service.html](http://www.cert.org/tech_tips/denial_of_service.html). Retrieved 10-3-2008.
3. Li, Wei. (lwei@nova.edu) Using Genetic Algorithm for Network Intrusion Detection. From <http://www.security.cse.msstate.edu/docs/Publications/wli/DOECSG2004.pdf>. Department of Computer Science and Engineering, Mississippi State University.
4. Rogers, L. (2004). "What is a Distributed Denial of Service (DDoS) Attack and What Can I Do About It?." From <http://www.cert.org/homeusers/ddos.html>. Retrieved 10-3-2008.
5. Compact Oxford English Dictionary of Current English. From [http://www.askoxford.com/concise\\_oed/genome?view=uk](http://www.askoxford.com/concise_oed/genome?view=uk). Retrieved 10-11-2008.
6. Sinclair, C., Pierce, L., & Matzner, S. (1999). "An Application of Machine Learning to Network Intrusion Detection." From <http://www.acsac.org/1999/papers/fri-b-1030-sinclair.pdf>. Retrieved 10-11-2008.
7. National Human Genome Research Institute Glossary. National Human Genome Research Institute - From <http://www.genome.gov/glossary.cfm?key=chromosome>. Retrieved 10-11-2008.
8. National Human Genome Research Institute Glossary. National Human Genome Research Institute. From <http://www.genome.gov/glossary.cfm?key=allele>. Retrieved 10-11-2008.
9. Marakas, George M. (2003). Modern Data Warehousing, Mining, and Visualization: Core Concepts. Upper Saddle River, NJ: Pearson Education. p. 142.
10. Ibid. p. 143.
11. Meffert, Klaus et al.: JGAP - Java Genetic Algorithms and Genetic Programming Package. URL: <http://jgap.sf.net>.

