NAME

findBgpAtoms – compute BGP policy atoms

SYNOPSIS

findBgpAtoms [-d debug-level] [-k] [-T dir] file-prefix

PARAMETERS

file-prefix

The prefix of filenames generated by straightenRV.

OPTIONS

–d *debug-level*

Verbosity of debugging output, 0 for no debugging output, 1 (default) for normal debugging output, 2 for all debugging output.

–k Whether to keep intermediate files or not after findBgpAtoms has run. All files generated by findBgpAtoms except file-prefix.atoms.ccdf, file-prefix.atoms.asp.gz, file-prefix.atoms.p2a and file-prefix.atoms.full.gz are intermediate files.

–T *dir* findBgpAtoms uses UNIX sort. This option provides sort with a temporary directory. Handy when sort runs out of space.

DESCRIPTION

findBgpAtoms computes BGP policy atoms. A BGP policy atom, or 'atom' for short, is defined relative to a system of BGP routers. This collection of scripts uses RouteViews peers as the system of BGP routers. Each atom consists of prefixes that are treated equivalently by the chosen system of BGP routers.

A precise definition of an atom is as follows. Two prefixes are said to be *path equivalent* if no BGP router can be found among the considered BGP routers that sees them with different AS paths. An equivalence class of this relation is called a *BGP policy atom*. It follows from this definition that prefixes in the same atom share a collection of AS paths.

Atoms are computed from a number of files generated by straightenRV, which must be located in the working directory:

file-prefix.pfasp.gz

The information needed to compute atoms: prefixes, peers and AS paths.

file-prefix.peer

Used for prefix selection. In this version, atoms are computed from 'global prefixes' only. (Non-global prefixes are ignored.) A global prefix is a prefix that is carried by all the peers listed in the file-prefix.peer file. Use straightenRV's -c option to create a selection of peers (and therefore global prefixes) based on peer size.

Many intermediate and auxiliary files are produced. The main output files for atoms are:

file-prefix.atoms.asp.gz

Maps each atom name to the collection of AS paths for the atom.

file-prefix.atoms.p2a

Maps each prefix to the atom that the prefix belongs to.

The following summarises the input and output files of straightenRV and findBgpAtoms: To summarise the above:

straightenRV -> .peer + .pfasp.gz + ...
.peer + .pfasp.gz -> findBgpAtoms -> .atoms.asp.gz + .atoms.p2a + ...

ATOM COMPUTATION

This section describes the computation of atoms by findBgpAtoms in several steps. Each step takes one or more files produced by the previous step and generates files to be used in the next step.

Sorting .pfasp.gz to .pfaspsorted

The .pfasp.gz file is sorted and written to a .pfaspsorted file. It sorts on prefix first, then on peer IP address, and finally on AS path.

Joining AS paths in .pfaspsorted to .pfaspcoll

For each prefix, groups all AS paths for that prefix in .pfaspsorted together, forming a collection of AS paths. Writes the prefix and the AS path collection to .pfaspcoll. Each AS path is first prepended with the IP address of the peer through which the AS path was learnt.

Only global prefixes are considered; non-global prefixes are dropped in this step, and are not written to .pfaspcoll. Nor are they considered in the remaining steps. See the 'file-prefix.peer' description above for a definition of global prefixes.

Sorting .pfaspcoll by AS path collection to .pfaspcollsorted

Sorts the .pfaspcoll file by the collections of AS paths. Note that this effectively places the prefixes that are in the same atom together.

Grouping prefixes in .pfaspcollsorted by atoms to .atoms

From the .pfaspcollsorted file, reads prefixes and their collections of AS paths, groups them into atoms, sorts the atoms by the number of prefixes per atom (in reverse order), and writes the result to the .atoms file. Also writes a distribution of prefix counts per atom to the .atoms.ccdf file.

The format of the .atoms file consists of four lines per atom:
1. The number of prefixes in the atom.
2. The sorted list of prefixes in the atom.
3. The collection of AS paths for the atom (as copied from .pfaspcollsorted).
4. An empty separator line.

Naming atoms in .atoms to .atoms.asp.gz, .atoms.p2a and .atoms.full.gz

Uniquely names the atoms found in .atoms. The atom name is a string composed as follows:

as<origin>np<#prefixes>at<sequence>

where <origin> is the origin AS of the atom (or multiple origin ASes separated by '_'), <#prefixes> is the number of prefixes in the atom, and <sequence> is a sequence number (starting with 1) ranging over all atom names that have the same <origin> and <#prefixes> parts. Examples:

as701np1133at1
as5676np125at1
as5676np125at2
as6140_16528np34at1

The following files are produced:

.atoms.asp.gz

Maps each atom name to the collection of AS paths for the atom. Note that in an earlier step, each AS path was prefixed with the IP address of the peer from which the AS path had been learnt.

.atoms.p2a

Maps each prefix to the atom that the prefix belongs to.

.atoms.full.gz

Contains the data of .atoms, adding an atom counter (counter ranging over all atoms found) and the atom name.

FILES

All output files generated by findBgpAtoms start with the *file-prefix* that was passed on the command line. The output files are described above.

SEE ALSO

straightenRV(1)

BUGS and TODO

Atoms definition

This script computes BGP policy atoms according to the definition of atoms given earlier. We should also compute atoms using a modified definition more appropriate for use within the 'atoms' project:

In determining path equivalence of two prefixes, we ignore the part of a prefix's AS path that does not include any of the RV peers' ASes. From this definition, it follows that prefixes in the same atom share a set of truncated AS paths, where each AS path is truncated to exclude the part that falls outside the set of peer ASes. Note that the number of atoms under this definition is smaller than the number of atoms under the original definition.

Origin-declared atoms

A third kind of atom exists, the 'origin-declared atom'. An origin-declared atom is defined by the AS that originates the prefixes in the atom. The prefixes in an origin-declared atom share the same origin link (the rightmost link in the AS path), i.e. are all announced by the same origin AS to the same neighbour AS. Origin-declared atoms are 'declared', in that they are 'imposed' by the origin AS; any policy applied by ASes other than the origin AS is ignored. Per prefix origin links are listed in the .pref file produced by straightenRV in the 'oriLkstats' column, but are not further grouped into atoms.

Statistics

Andre Broido's original findBgpAtoms script produced many interesting statistics on atoms that will one day be incorporated.

AUTHORS

Patrick Verkaik (patrick@caida.org)
Andre Broido: algorithms and scripts before rewrite
Young Hyun: code review

REFERENCES

Atoms web page:
http://www.caida.org/projects/routing/atoms/

Andre Broido, kc claffy, ‘Analysis of RouteViews BGP data: policy atoms’, Proceedings of the Network-

Related Data Management workshop, Santa Barbara, May 23, 2001.
http://www.caida.org/outreach/papers/2001/NdrmBgp/

Andre Broido, Evi Nemeth, kc claffy, ‘Internet Expansion, Refinement, and Churn’, European Transactions on Telecommunications, January 2002.
http://www.caida.org/outreach/papers/2002/EGR/

Andre Broido, kc claffy, ‘Complexity of global routing policies’.
http://www.caida.org/outreach/papers/2001/CGR/