This document describes the process of installing Bind 9.x on your Linux box as a Caching DNS server.

The steps to install it are as follows:

1. [Install](http://linux-sxs.org/security/openssl.html) Openssl
2. Download the latest stable release from [ISC.org](http://www.isc.org/products/BIND/bind9.html) \*
3. Extract the tarball like so:
   * tar zxvf bind-9.tar.gz
   * cd bind-9.x
4. Configure the software:
   * ./configure --prefix=/usr \
   * --sysconfdir=/etc \
   * --enable-threads \
   * --localstatedir=/var/state \
   * --with-libtool \
   * --with-openssl=/usr/ssl
5. Compile it:
   * make
6. Remove all existing Bind software:
   * rpm -q -a | grep '^bind' | while read line
   * do
   * rpm -e --nodeps $line
   * done
7. Install your new Bind:
   * make install
   * cd doc/man/bin (not needed on 9.2.0 and above)
   * for i in 1 5 8 (not needed on 9.2.0 and above)
   * do (not needed on 9.2.0 and above)
   * install \*.$i /usr/man/man$i (not needed on 9.2.0 and above)
   * done (not needed on 9.2.0 and above)
   * cd ../dnssec (not needed on 9.2.0 and above)
   * install \*.8 /usr/man/man8 (not needed on 9.2.0 and above)
8. Update your library resolutions:
   * ldconfig -v
9. Create the Bind user and group
   * groupadd named
   * useradd -d /var/named -g named -s /bin/false named
10. Adjust the group/perms on /var/run
    * vigr (add named to the 'daemon' group)
    * chown root:daemon /var/run
    * chmod 775 /var/run
11. Create the Bind rundir
    * mkdir -p /var/named/pz
    * chown -R named:named /var/named
    * chmod -R 755 /var/named
12. Create a script to maintain the root.hints file
    * cat << "EOF" > update\_named
    * #!/bin/sh
    * cd /var/named
    * wget http://dns.vrx.net/tech/rootzone/db.root
    * if [ -s /var/named/db.root ] ; then
    * chown named:named /var/named/db.root
    * /etc/rc.d/named stop
    * mv /var/named/root.hints /var/named/root.hints.old
    * mv /var/named/db.root /var/named/root.hints
    * /etc/rc.d/named start
    * fi
    * EOF
13. Make the script executable, and execute it (Bind will probably fail, but your root.hints file will get updated like we wanted)
    * chmod 700 update\_named
    * ./update\_named
14. Move the script to your monthly cron directory
    * mv update\_named /etc/cron.monthly
15. Create /var/named/pz/127.0.0 as below,
16. $TTL 1D
17. @ 1D IN SOA localhost. root.localhost. (
18. 42 ; serial (d. adams)
19. 3H ; refresh
20. 15M ; retry
21. 1W ; expiry
22. 1D ) ; minimum
23. 1D IN NS localhost.
24. 1 1D IN PTR localhost.
25. Create /var/named/pz/192.168.1
    * ln -s 127.0.0 192.168.1
26. Create /etc/resolv.conf
    * echo "nameserver 127.0.0.1" > /etc/resolv.conf
27. Create your rndc password (we'll use "hush" for ours)
    * mmencode (this command is part of the metamail package)
    * hush
    * aHVz (mmencode returns this)
    * ^C
28. Create /etc/rndc.conf
29. // this file is used by the rndc utility
30. options {
31. // what host should rndc attempt to control by default
32. default-server localhost;
33. // and what key should it use to communicate with named
34. default-key "rndc-key";
35. };
36. server localhost {
37. // always use this key with this host
38. key "rndc-key";
39. };
40. key "rndc-key" {
41. // how was the key encoded
42. algorithm hmac-md5;
43. // what's the password
44. secret "aHVz";
45. };
46. // secret was generated by running mmencode on command line
47. // and then entering a secret phrase
49. Create /etc/rndc.key
50. // this file is used when named starts up and sees that
51. // there is a key assigned to the control channel
52. key "rndc-key" {
53. // how was the key encoded
54. algorithm hmac-md5;
55. // what's the password
56. secret "aHVz" ;
57. };
59. And finally, create /etc/named.conf as below
60. // This is a configuration file for named (from BIND 9.0 or later).
61. // It would normally be installed as /etc/named.conf.
62. //
63. // Changed to match secure example from LASG 5/17/00
64. // Changed to match Linux Journal example 9/17/00
65. // Added new "view' sections to stop fingerprinting of Bind 9.x per
66. // Bugtraq 1/31/00
67. // Added rndc key stuff per DNS & Bind (Rev. 4) Chapter 11
68. // added use-id-pool and more comments based on above chapter
69. options {
70. // Directory where bind should create files if
71. // not explicitly stated
72. directory "/var/named";
73. // whom do we allow to do zone tranfers
74. allow-transfer { 192.168.1.0/24; };
75. // new in Bind 9.x to allow RFC1886 -> RFC2874 conversion
76. // to support IPv6
77. // allow-v6-synthesis { 192.168.1.10; };
78. // OBSOLETED in 9.3.0 + !!
79. // tell Bind to check the names in zone files
80. // since it no longer does this by default
81. // (unimplemented 9.3.0+)
82. check-names master warn;
83. // sets the size of something or other to 20Mb ;)
84. datasize 20M;
85. // sets the size of the journal to 5Mb
86. max-journal-size 5M;
87. // Bind 9.x doesn't recognize this yet :(
88. // deallocate-on-exit no;
89. // where should Bind put a dump of its cache
90. // if told to dump it
91. dump-file "named\_dump.db";
92. // how often should bind check for new
93. // interfaces toi listen on. we turn
94. // this off by setting it to 0
95. interface-interval 0;
96. // specify what interfaces/ips to listen on
97. // as the default is all of them
98. listen-on { 192.168.1.10; 127.0.0.1; };
99. // define a mximum size of cached records
100. // new in Bind 9.x
101. max-cache-size 20M;
102. // where to right stats of memory usage
103. // Bind 9.x doesn't recognize this yet :(
104. memstatistics-file "named.memstats";
105. // where to put out pid file
106. // absolute path since we don't want
107. // it in /var/named
108. pid-file "/var/run/named.pid";
109. // force Bind to use port 53 for its
110. // network operation to other DNS
111. // servers (Bind 9 uses high ports
112. // by default). Makes firewalling easier
113. query-source address \* port 53;
114. transfer-source \* port 53;
115. notify-source \* port 53;
116. // where to dump Bind server stats
117. statistics-file "named.stats";
118. // force Bind to be "more" random in assiging
119. // message ids
120. use-id-pool yes;
121. // If the chaos view below doesn't work
122. // for some reason, still give out a bogus
123. // answer for Bind version requests
124. version "This is not the port you're looking for.";
125. // keep stats on a zone basis
126. zone-statistics yes;
127. };
128. controls {
129. // this allows rndc to be used from the localhost
130. // to talk to bind on the loopback interface
131. // using the key defined as 'rndc-key'
132. inet 127.0.0.1 allow { localhost; } keys { rndc-key; };
133. };
134. // the rest of the key configuration is in
135. // /etc/rndc.conf and the key itself is in
136. // /etc/rndc.key
137. key "rndc-key" {
138. // how was key encoded
139. algorithm hmac-md5;
140. // what is the pass-phrase for the key
141. secret "aHVz" ;
142. };
143. logging {
144. channel named\_info {
145. // log to syslog instead of a file
146. syslog;
147. // include the category of the event in the log
148. print-category yes;
149. // include the severity of the event in the log
150. print-severity yes;
151. // include the time of the event in the log
152. print-time yes;
153. };
154. // Processing of client requests
155. category client { named\_info; };
156. // named.conf parsing and processing
157. category config { named\_info; };
158. // Messages relating to internal memory structures
159. category database { named\_info; };
160. // This is the default for any category not specifically defined
161. category default { named\_info; };
162. // The catch-all. Anything without a category of its own
163. category general { named\_info; };
164. // Uncomment if you dont want to know about lame server.
165. // Leave commented and it defaults to the
166. // value of default above
167. // category lame-servers { null; };
168. // The NOTIFY protocol
169. category notify { named\_info; };
170. // Network operations
171. category network { named\_info; };
172. // DNS resolution like recursive lookups, etc..
173. category resolver { named\_info; };
174. // Approval and denial of requests
175. category security { named\_info; };
176. // Dynamic updates
177. category update { named\_info; };
178. // Queries. Duh.
179. category queries { named\_info; };
180. // Zone transfers received
181. category xfer-in { named\_info; };
182. // Zone transfers sent
183. category xfer-out { named\_info; };
184. };
185. // this is where we define different versions
186. // of our zones based on where the client is
187. // coming from.
188. // the first view that matches a client is
189. // the one that gets used, so order can be
190. // important
191. view "external-chaos" chaos {
192. // you could use 'any' or even 'localnets' here
193. // instead of specifying each IP range
194. // however, it should be noted that 'localnets'
195. // means ANY network Bind is directly connected
196. // to which might include your ISP
197. match-clients { 192.168.1.0/24; 127/8; };
198. recursion no;
199. zone "." {
200. type hint;
201. // this causes a null response to queries
202. // about the Bind version
203. file "/dev/null";
204. };
205. };
207. view "external" {
208. // you could use 'any' or even 'localnets' here
209. // instead of specifying each IP range
210. // however, it should be noted that 'localnets'
211. // means ANY network Bind is directly connected
212. // to which might include your ISP
213. match-clients { 192.168.1.0/24; 127/8; };
214. zone "." {
215. type hint;
216. file "root.hints";
217. };
218. };
220. view "external-127" {
221. // you could use 'any' or even 'localnets' here
222. // instead of specifying each IP range
223. // however, it should be noted that 'localnets'
224. // means ANY network Bind is directly connected
225. // to which might include your ISP
226. match-clients { 192.168.1.0/24; 127/8; };
227. zone "0.0.127.in-addr.arpa" {
228. type master;
229. file "pz/127.0.0";
230. allow-update {
231. none;
232. };
233. };
234. };
236. view "external-192" {
237. // you could use 'any' or even 'localnets' here
238. // instead of specifying each IP range
239. // however, it should be noted that 'localnets'
240. // means ANY network Bind is directly connected
241. // to which might include your ISP
242. match-clients { 192.168.1.0/24; 127/8; };
243. zone "1.168.192.in-addr.arpa" {
244. type master;
245. file "pz/192.168.1";
246. allow-update {
247. none;
248. };
249. };
250. };
252. The only thing left to do is start Bind:
     * /usr/sbin/named -u named

Congrats! You now have a fairly secure, caching name server that can be controlled using rndc!

Enjoy your new Bind server!