

```
# Copyright 2011–2012 Amazon.com, Inc. or its affiliates. All Rights Reserved.
#
# Licensed under the Apache License, Version 2.0 (the "License"). You
# may not use this file except in compliance with the License. A copy of
# the License is located at
#
# http://aws.amazon.com/apache2.0/
#
# or in the "license" file accompanying this file. This file is
# distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF
# ANY KIND, either express or implied. See the License for the specific
# language governing permissions and limitations under the License.
```

```
module AWS
  class EC2
```

```
    class VPNGatewayCollection < Collection
```

```
      include TaggedCollection
      include Core::Collection::Simple
```

```
      # Creates a new virtual private gateway. A virtual private gateway is
      # the VPC-side endpoint for your VPN connection. You can create a
      # virtual private gateway before creating the VPC itself.
```

```
      #
      # @param [Hash] options
      #
      # @option options [String] :vpn_type ('ipsec.1') The type of VPN
      #   connection this virtual private gateway supports.
      #
      # @option options [AvailabilityZone,String] :availability_zone
      #   The Availability Zone where you want the virtual private gateway.
      #   AWS can select a default zone for you. This can be an
      #   {AvailabilityZone} object or availability zone name string.
```

```
      #
      # @return [VPNGateway]
```

```
      #
      def create options = {}
```

```
        client_opts = {}
        client_opts[:type] = options[:vpn_type] || 'ipsec.1'
```

```
        if az = options[:availability_zone]
          az = az.name if az.is_a?(AvailabilityZone)
          client_opts[:availability_zone] = az
        end
```

```
        resp = client.create_vpn_gateway(client_opts)
```

```
        VPNGateway.new_from(:create_vpn_gateway, resp.vpn_gateway,
          resp.vpn_gateway.vpn_gateway_id, :config => config)
```

```
      end
```

```
      # @param [String] vpn_gateway_id
```

```
      # @return [VPNGateway]
```

```
      def [] vpn_gateway_id
```

```
        VPNGateway.new(vpn_gateway_id, :config => config)
```

```
      end
```

```
protected

def _each_item options = {}, &block
  response = filtered_request(:describe_vpn_gateways, options, &block)
  response.vpn_gateway_set.each do |g|

    gateway = VPNGateway.new_from(:describe_vpn_gateways, g,
      g.vpn_gateway_id, :config => config)

    yield(gateway)
  end
end

end

end

end
```